27. Inventory Logistical Management (ILM)

27.1 Remedy-ILM (Inventory, Logistics and Maintenance {ILM} Manager)

ILM helps the M&O staffs at the EDF and DAACs maintain records that describe all inventory items, as well as their EIN structures, repair histories, and locations. The system keeps chronological histories (a record of the transactions) of installation, relocation, movement, shipment and archiving of inventory items. ILM is used by the Property Management, Maintenance, and Logistics teams in managing the tangible property of NASA's EOSDIS project. In addition, this system also manages COTS software license information, such as the Rights-to-use purchased, allocated, and remaining.

ILM is a customized application of the Remedy Action Request System (ARS). The customizations adapt the product to the ILS processes used for ECS. ILM takes into account how business rules and logistics concepts are applied on the ECS project. This document does not address these considerations in detail, but the following general introduction should help.

Each inventory item is identified by a unique Equipment Inventory Number (EIN). In the case of hardware items, an EIN corresponds to a silver sticker affixed to the item. Some of the items are shipped to sites and installed. Others such as consumables are issued but not installed. After a period, some items may be transferred to other locations or relocated for use with other parent machines. Items are archived when no longer needed or serviceable.

For tracking and auditing purposes, inventory items – especially hardware – are allocated to ECS "parent" machines. These parent and child relationships are called an EIN structure. EIN structures have active and inactive dates that establish the timeframe during which the pairing is in effect.

Table 27.1-1 summarizes the operator functions that Remedy ILM supports, and table 27.1-2 describes Remedy-ILM user groups and what privileges each group have. The sections that follow present how to use Remedy features that were customized for ECS inventory, logistics, and maintenance management. In addition, these sections also have detail instructions on how to perform functions designed for each individual form. For more information on Remedy's Action Request System, refer to Remedy help manual.

611-EMD-001

Table 27.1-1. Common ECS Operator Functions Performed with ILM

Operating Function	GUI	Section	Description	When and Why to Use
Remedy Overview	Navigating Remedy User Tool Defining Search Criteria ILM Predefined Reports	27.2.1 27.2.2 27.2.3	Overview of Remedy User Tool.	To Navigate, search, and run ad hoc and predefined reports.
Property Management	ILM-EIN ILM-EIN Structure ILM-EINTransactions ILM-Transaction Log ILM-OEM Parts ILM-Vendor-MFR ILM-HwSw Code ILM-Status Codes ILM-Maint Contract ILM-Sites ILM-Inventory Location	27.3.1 27.3.2 27.3.3 27.3.4 27.3.5 27.3.6 27.3.7 27.3.8 27.3.9 27.3.10 27.3.11	Maintain information about accountable property items, their product structures, and inter-relationships.	To maintain information that specifies the identity, source, location, transfer, relocation, and installation of procured inventory items.
Property Maintenance	ILM-MWO ILM-MWO Line Item	27.4.1 27.4.2	Manage information for required maintenance repairs.	To track and monitor maintenance activities
License Management	ILM-License Product ILM-License Entitlement ILM-License ILM-License Mapping ILM-Additional Host	27.5.1 27.5.2 27.5.3 27.5.4 27.5.5	Manage entitlements, licenses, and license allocations for licensed COTS software.	To track the receipt, movement, and consumption of software licenses and their associated rights-to-use.
System Administrator	ILM-System Parameters Intersite Date Exchange User Remedy Admin Tool Database Special Constraints Event and Error Messages	27.6.1 27.6.2 to 27.6.4 27.6.5 27.6.6 27.6.7 27.6.8 27.6.9	Manage AR System	Uses by Remedy Administrator to administrate Remedy Tool.

Table 27.1-2. Remedy-ILM Groups Description (1 of 3)

Group Name	Description	Access
ILMAdmin	Full privileges to all operator and system administrator functions within ILM	ALL ILM related Forms
ILMProperty	To maintain information that specifies the identity, source, location, transfer, relocation, and installation of procured inventory items.	♦ ILM-EIN (**Change**) ♦ ILM-EIN Transactions (**Change**) ♦ ILM-Transaction Log (**View**) ♦ ILM-Vendor-MFR (**Change**) ♦ ILM-HwSw Code (**View**) ♦ ILM-OEM Parts (**Change**) ♦ ILM-Status Codes (**View**) ♦ ILM-Maint Contract (**View**) ♦ ILM-Sites (**View**) ♦ ILM-Inventory Location (**View**) ♦ ILM-MWO (**Change**) ♦ ILM-MWO Line Item (**Change**) ♦ ILM-License Product (**View**) ♦ ILM-License Entitlement (**View**) ♦ ILM-License (**View**) ♦ ILM-License Mapping (**View**) ♦ ILM-Additional Host (**View**)
ILMContract	Maintains maintenance contract purchased for ECS Hardware and COTS software.	♦ ILM-EIN (**Change - only to the Maint Contract ID, Maint Exp Date, and Maint Vendor on this form**) ♦ ILM-EIN Transactions (**View**) ♦ ILM-Transaction Log (**View**) ♦ ILM-Vendor-MFR (**View**) ♦ ILM-HwSw Code (**View**) ♦ ILM-OEM Parts (**View**) ♦ ILM-Status Codes (**View**) ♦ ILM-Maint Contract (**Change**) ♦ ILM-Sites (**View**) ♦ ILM-Inventory Location (**View**) ♦ ILM-MWO (**View**) ♦ ILM-MWO Line Item (**View**) ♦ ILM-License Product (**View**) ♦ ILM-License Entitlement (**Change - only to the Maint Contract ID and Maint Exp Date on this form**) ♦ ILM-License (**View**) ♦ ILM-License Mapping (**View**) ♦ ILM-Additional Host (**View**)

Table 27.1-2. Remedy-ILM Groups Description (2 of 3)

Group Name	Description	Access
ILMlicuser	License management data update privileges for software license administrators	♦ ILM-EIN (**View**) ♦ ILM-EIN Transactions (**View**) ♦ ILM-Transaction Log (**View**) ♦ ILM-Vendor-MFR (**View**) ♦ ILM-HwSw Code (**View**) ♦ ILM-OEM Parts (**View**) ♦ ILM-Status Codes (**View**) ♦ ILM-Maint Contract (**View**) ♦ ILM-Sites (**View**) ♦ ILM-Inventory Location (**View**) ♦ ILM-MWO (**View**) ♦ ILM-MWO Line Item (**View**) ♦ ILM-License Product (**Change**) ♦ ILM-License Entitlement (**Change**) ♦ ILM-License (**Change**) ♦ ILM-License Mapping (**Change**) ♦ ILM-Additional Host (**Change**)
ILMmaint	Maintenance management data update privileges for EDF's Maintenance Coordinator	♦ ILM-EIN (**View**) ♦ ILM-EIN Transactions (**View**) ♦ ILM-Transaction Log (**View**) ♦ ILM-Vendor-MFR (**View**) ♦ ILM-HwSw Code (**View**) ♦ ILM-OEM Parts (**View**) ♦ ILM-Status Codes (**View**) ♦ ILM-Maint Contract (**View**) ♦ ILM-Sites (**View**) ♦ ILM-Inventory Location (**View**) ♦ ILM-MWO (**Change**) ♦ ILM-MWO Line Item (**Change**) ♦ ILM-License Product (**View**) ♦ ILM-License Entitlement (**View**) ♦ ILM-License (**View**) ♦ ILM-License Mapping (**View**) ♦ ILM-Additional Host (**View**)
ILMdaacAdmin	Full privileges to all operator and system administrator functions within ILM at a site.	ALL Remedy-ILM related forms.

Table 27.1-2. Remedy-ILM Groups Description (3 of 3)

Group Name	Description	Access
ILMdaacMaint	Maintenance management data update privileges for Local Maintenance Coordinator at the site.	♦ ILM-EIN (**View**) ♦ ILM-EIN Transactions (**View**) ♦ ILM-Transaction Log (**View**) ♦ ILM-Vendor-MFR (**View**) ♦ ILM-HwSw Code (**View**) ♦ ILM-OEM Parts (**View**) ♦ ILM-Status Codes (**View**) ♦ ILM-Maint Contract (**View**) ♦ ILM-Sites (**View**) ♦ ILM-Inventory Location (**View**) ♦ ILM-MWO (**Change**) ♦ ILM-MWO Line Item (**Change**) ♦ ILM-License Product (**View**) ♦ ILM-License Entitlement (**View**) ♦ ILM-License (**View**) ♦ ILM-License Mapping (**View**) ♦ ILM-Additional Host (**View**)
ILMquery	ILM data query privileges only.	♦ ILM-EIN (**View**) ♦ ILM-EIN Transactions (**View**) ♦ ILM-Transaction Log (**View**) ♦ ILM-Vendor-MFR (**View**) ♦ ILM-HwSw Code (**View**) ♦ ILM-OEM Parts (**View**) ♦ ILM-Status Codes (**View**) ♦ ILM-Maint Contract (**View**) ♦ ILM-Sites (**View**) ♦ ILM-Inventory Location (**View**) ♦ ILM-MWO (**View**) ♦ ILM-MWO Line Item (**View**) ♦ ILM-License Product (**View**) ♦ ILM-License Entitlement (**View**) ♦ ILM-License (**View**) ♦ ILM-License Mapping (**View**) ♦ ILM-Additional Host (**View**)

611-EMD-001

27.2 Remedy User Tool Overview

27.2.1   Navigating Remedy User Tool

Invoking Remedy-ILM from a PC

To start Remedy User,

- Click Start →Programs →Action Request System →Remedy User

- Double-click on a Remedy User icon on the desktop, if one exists.

The Remedy User screen displays. Enter user Id and password.

Once logged into Remedy User, user can open a form. To view a list of all available forms, select **File → Open,** or select the Open icon, the first icon in the toolbar. This display the complete list of forms to which the user have access (see Figure 27.2.1-1).
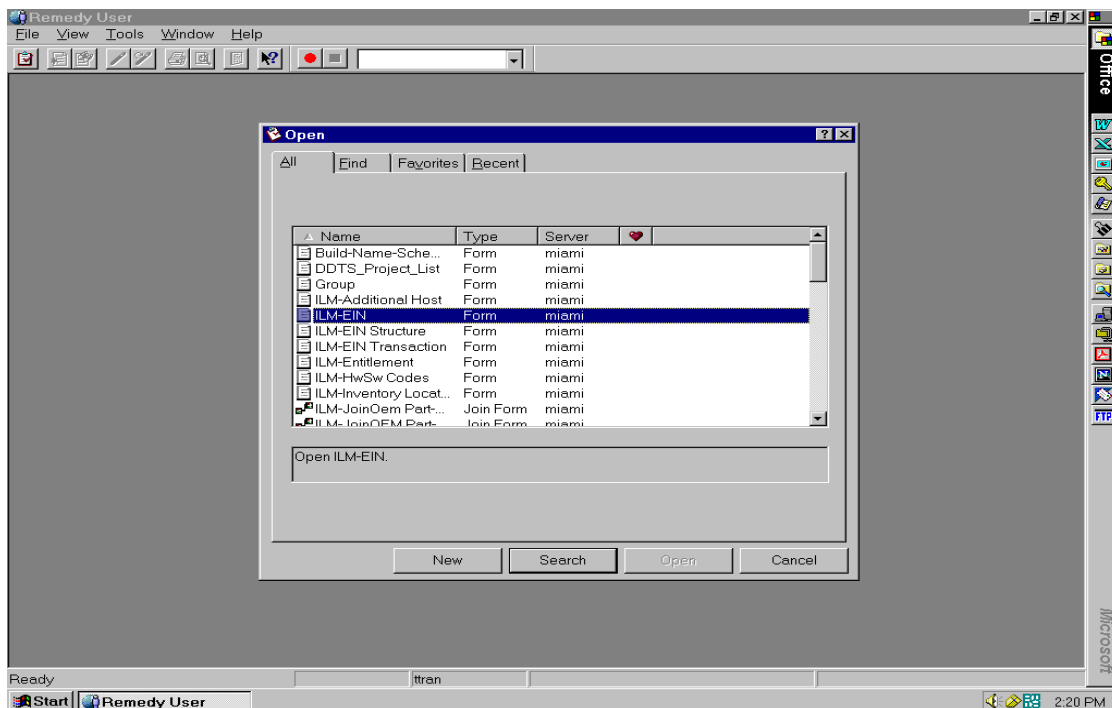
Figure 27.2.1-1. Open GUI

♦ To view a shorter list of forms, select the **Find, Favorites, and Recent** tabs in the Open dialog box.

   ♦ **Find** – Lists only the requested forms.
   ♦ **Favorites** – Lists only the forms designates as favorites.

♦ **Recent** – Lists only the recently used forms.

♦ Remedy application is a Graphical User Interface (GUI). The GUI is very similar to any Microsoft application. It has a title bar, menu bar, tool bar, and status bar.

- ♦ Title Bar – Displays form name and mode.

- ♦ Menu Bar – The menus File, Edit, View, Tools, Actions, Window, and Help are stardard AR System menus. Any other menus are those customized for a specific form. For instant, the ILM-EIN form has an additional menu selection calls ILM Reports.

- ♦ Tool Bar – Displays Remedy tool bar icons.

- ♦ Status Bar – Displays form status, logged on user and server name

♦ Remedy application is case sensitve. The convention in Remedy-**ILM** is that most of the data are in UPPER CASE One exception is **ECS Name** which holds a host's name. To follow ECS's naming convention, some of the ECS Name will be in lower case and some will be in upper case.

♦ Pressing the <**ENTER**> key after each entry **DOES NOT WORK.** Use either the <TAB> key or the mouse to move to the next field.

♦ Each form can be open in one of two modes:

- ♦ **New** – To create a new record

- ♦ **Search** – To search for an existing record(s).

- ♦ An easy way to change a form's mode to New or Search is to click the **New Request** or **New Search** icon in the toolbar.

♦ Field Labels Convention

- ♦ **Required Fields** – required fields have labels in **bold type**. Except in cases where field values are added by default or automation, user must enter information in a required field. Otherwise, an attempt to save the request is rejected by the AR system.

- ♦ **Optional Fields** –Optional fields have labels in regular type. Though this information is not always required to save the record, optional data helps to resolve, track, and reference a record.

- ♦ *System-generated Fields* – System-generated fields have labels in *italicized type*. User does not enter information in these fields when saving a request, as information in these fields is automatically generated by the system.

27.2.2   Defining Search Criteria

27.2.2.1   Searching by Example

The easiest way to perform a search is to have a form in Search mode, fill one or more specific field values to search for, then click search. Remedy performs the search and displays the matching records. User may specify values for as many fields as needed. The more fields are filled, the more specific the search becomes. Data specified as search criteria is treated as a logical AND. Figure 27.2.2.1-1 displays an example of a search by example.   This search will select all records where the manufacturer is SGI, AND the hardware and software code is Hardware (H), AND the Location is GSFC. Table 27.2.2.1-1 describes relational operators that can be used in the query window for Search by Example.

Figure 27.2.2.1-1. Search by Example

Table 27.2.2.1-1. Relational Operators Used in the Query Window

Operator	Action
<	Matches contents that are *less than* the value.
>	Matches contents that are *greater than* the value.
!=	Matches centents that are *not equal* to the value.
<=	Matches centents that are *less than or equal* to the value.
>=	Matches centents that are *greater than or equal* to the value.
=	Matches centents that are *equal* to the value.

Note: Stopping a Search – While the system is performing the search, user can stop the search by clicking **Stop Search** in the Active Search dialog.

27.2.2.2 Using the Advanced Search Bar

Using the Advanced Search Bar is a second method of specifying search criteria.

To display the Advanced Search Bar, click on the **Advanced** button in the search form banner, or choose **View -> Advanced Search Bar.** The bar displays at the bottom of the form. With the Advanced Search Bar method, the search criteria are defined in a statement, rather than in field value examples. The basic syntax for building a statement is:

Field_Label   Relational_Operation   Field_Value

♦ Field_Label must be enclosed with single quote (**'**).

♦ Field_Value must be enclosed with double quote (**"**).

1. **Enter a Field Label**

   Use one of the following ways to enter a field label in the Advanced Search Bar:

   ♦ Click on a field label in the form. The field label displays in the Advanced Search Bar, enclosed in single quotes.

   ♦ Select the field name from the **Fields** menu. The name displays in the Advanced Search Bar, enclosed in single quotes.

   ♦ In the Advanced Search Bar, type the field label between single quotes.

2. **Add a Relational Operator**

   ♦ Click on the appropriate operator from the palette of buttons in the Advanced Search Bar area, or type the operator. Table 27.2.2.2-1 lists and describes the action each operator does. Table 27.2.2.2-2 describes the use of wildcard symbols.

3. **Enter a Field Value**

   The field value can be text, a number, or a keyword (Table 27.2.2.2-3 defines some of Remedy Keywords definition). To add a field value in the Advanced Search Bar, use one of the following methods:

♦ Type the field value in the Advanced Search Bar enclosed the value with double quotes.

♦ Select the field value from the **Fields -> Selection Values** menu.

4. **Click on the Search icon or button to start the search.**

Once the search criteria are specified, user may display, modify, or generate reports from the results of the search.

Figure 27.2.2.2-1 illustrates how to use the Advanced Search Bar. This example selects all records where ECS Name contains the word "MOP" anywhere in the ECS Name and location equal to EMOSG.

Figure 27.2.2.2-1. Using the Advance Search Bar

Table 27.2.2.2-1. Operators Used in the Advanced Search Bar

Operator	Action
AND **&&**	Logical AND of the result of two conditions (the result is true only if both conditions are true). For example, 'MFR' = "SGI" AND 'Location' = "PVC" finds all SGI's equipment locates in PVC. You can use the symbol && instead of the word AND.
OR ‖	Logical OR of the result of two conditions (the result is true if either conidtion is true). For example, 'MFR' = "SGI" OR 'MFR' = "SUN" finds all SGI and SUN equipment. You can use the symbol ‖ instead of the word OR.
NOT **!**	Negates the condition that follows (if the condition is false, the result is true). For example, NOT 'Room' = "1073" finds all entries that are not in room 1073. You can use the symbol ! instead of the word NOT.
LIKE	Performs a pattern search. For example, 'ECS Name' LIKE "g0%" finds all entries where the ECS Name starting with g0.
+	♦ Adds two integer or real values. ♦ Adds an integer interval to a time value. ♦ Concatenates two character strings.
-	♦ Subtracts two integer or real values. ♦ Subtracts two time values. ♦ Subtracts an integer interval from a time value.
*****	Multiples two integer or real values.
/	Divides two integer or real values.
%	Supplies the modulo of two integer values (the remainder of a division of the values).
<	Matches contents that are less than the value.
>	Matches contents that are greater than the value.
!=	Matches centents that are not equal to the value.
<=	Matches centents that are less than or equal to the value.
>=	Matches centents that are greater than or equal to the value.
=	Matches centents that are equal to the value.

Table 27.2.2.2-2. Wildcard Symbols

Wildcard	Action
%	Matches any string of 0 or more characters. For example, 'ECS Name' = "%dms%" matches all ECS Name having dms anywhere in the ECS Name.
_	(Underbar). Matches any single character. For example, B_b matches Bab, Bob, and Bub.
-	(Hyphen). Indicates a range. Always use within brackets (**[]**).
[]	Use to match any single character within a specified range or set. For example, **[a-f]** matches the range of characters a throught f while **[abcf]** matches the set of characters a, b, c, or f.
[^]	Matches any single character *not* within a specified range or set. For example, **[^a-f]** matches all characters except the range a through f while **[^abcf]** matches all characters except **a,b,c** or **f.**

Table 27.2.2.2-3. Using Keywords

Keyword	Action
$DATE$	Current date.
$TIME$	Current time.
$TIMESTAMP$	Current date and time.
$NULL$	Name of the user who is currently logged in.
$USER$	Lacking a value.

27.2.3 ILM Predefined Reports

ILM Predefined reports are reports that provide specific ILM information. Table 27.2.3-1 lists and defines these reports. This table also list the section numbers where the instruction to generate these reports can be found.

Table 27.2.3-1. ILM Pre-Defined Reports (1 of 2)

Report Type	Report Description	Table No.
Inventory Management		
Install/Receipt Report	A receipt describing an operator-specified EIN item together with all of its associated components order by EIN number.	27.3.1.1-2
Installation Report	A receipt describing an operator-specified EIN item together with its components having status "I" (for installed).	27.3.1.1-2
Parent EIN Report	Provides a listing of only Parent items.	27.3.1.1-2
Parent EIN and total System Cost Report	Provides a listing of only Parent items and the total system cost for each Parent.	27.3.1.1-2
Inventory Report	Provides an ASCII formated report identifying the inventory items by Parent EIN according to the operator-specified criteria.	27.3.1.1-2
Quarterly Property Management Report	Provides a list of contractor-acquired equipment items by quarter, sorted by Mfr and product description.	27.3.1.1-2
Purchase Order Cost Report	Provides a list of EINs and their cost associated with an operator-specified purchase order.	27.3.1.1-2
Cost – Selected ECS Managed Property	Provides the quantity and total cost of operator-selected EINs, grouped by type of inventory (Hardware, Software, Consumable, i.e.).	27.3.1.1-2
EIN Transaction History	A list of the transactions processed for operator-specified items during an operator-specified timeframe, sorted by EIN number and "from" location	27.3.1.1-2
Spare Equipment	A list of spare equipment sorted by EIN number.	27.3.1.1-2
ECS Shipment Report	Provides a listing of items that were shipped within an operator-specified time frame.	27.3.4-2
Maintenance Management		
Maintenance Work Order Verification Report	A full description of operator-selected work orders and the items undergoing maintenance action that they cover.	27.4.1-3
Maintenance Contract Report	Provides a list of operator-specified maintenance contract and all the associated items the contract covers.	27.3.9-2
RMA Work Order Report	Provides an ASCII formated spreadsheet formated report with embedded formulas for RMA data.	27.4.1-3

Table 27.2.3-1. ILM Pre-Defined Reports (2 of 2)

Report Type	Report Description	Table No.
License Management		
License Entitlements Status Report	Lists the status of current license entitlements for licensed software products, sorted by software product, version, and license type.	27.5.2-2
License Allocations by Product Report	Lists license allocations for licensed software products, sorted by product, version, and host name.	27.5.3-3
License Allocations by Host Report	Lists license allocations, sorted by host name and ECS part alias.	27.5.3-3

27.3 Property Management

Remedy provides the M&O staffs at the EDF and the DAACs the capability to maintain inventory records, including EIN structures. Property Administrators can submit new records, modify existing ones, and perform transactions that capture installation, relocation, movement, shipment and archive activities. These transactions are logged for historical purposes. The following sections will describe these functions in detail.

27.3.1   ILM-EIN GUI

The ILM-EIN form (Figures 27.3.1-1 through 27.3.1-5) is used for creating, viewing or modifying all ECS inventory records. In addition, this form also allows the Property Administrator to create and modify EIN structures via the Parent EIN field. Other ILM groups may view and perform reports on this form. The following tables provide detail instructions to perform the following functions:

- ♦ Adding New Inventory Item (Table 27.3.1-2)

- ♦ Adding New Component to an EIN structure (Table 27.3.1-3)

- ♦ Modifying EIN record (Table 27.3.1-4)

- ♦ Run ILM-Predefined reports (Table 27.3.1-5)

Figure 27.3.1-1. ILM-EIN (Part Info and Location & Purchasing Info) GUI

Figure 27.3.1-2. ILM-EIN (Maintenance & Other Info.) GUI

Figure 27.3.1-3. ILM-EIN (Components) GUI

Figure 27.3.1-4. ILM-EIN (Maintenance Contract) GUI

Figure 27.3.1-5. ILM-EIN (History) GUI

Table 27.3.1-1 describes the fields on the ILM- EIN form.

Table 27.3.1-1. ILM-EIN Form Field Description (1 of 2)

Field Name	Data Type	Size	Entry	Description
EIN	Char	20	Optional	Identifier for an inventory item.
Parent EIN	Char	20	Optional	EIN of the host of which this item is a component of.
ECS Name	Char	30	Optional	Name of the machine with which the item is associated.
Serial No	Char	30	Optional	Manufacturer's serial number of the item.
Part No	Char	34	Optional	Manufacturer's or vendor's part number.
Description	Char	60	Optional	Manufacturer's or vendor's description for the item.
Hw-Sw Code	Char	2	Optional	Code for classifying inventory items by type.
MFR	Char	6	Optional	Code used for the manufacturer.
Mod-Ver	Char	24	Optional	Model or version of the item.
Location & Purchasing Info.			Page	Contains the following fields about the EIN item: Location, Building, Room, Item Status, Vendor ID, PO Number, Cost, Quantity, Receive Date, Installation date, and Audit Date.
Location	Char	6	Optional	Identifier that designates an inventory location.
Building	Char	6	Optional	Identifier for the building where the item can be found.
Room	Char	15	Optional	Identifier for the room where the item can be found.
Item Status	Char	2	Optional, default R.	Code that designates the status of the item. The following values are set when processing transactions: R = Received; SP = Spare Equipment; I = Installed; X = Returned to vendor; G = Transfer to Gov; TV = Trade in to Vendor; S = Excess sold to vendor; RG = Relieved from accountability
Vendor ID	Char	6	Required	Code for the Vendor from whom the item was purchased.
PO Number	Char	10	Required	Identifier of the purchase order against which the item was received.
Cost	Decimal	10.2	Optional	Purchase cost of the item.
Quantity	Integer	4	Optional	Number of items purchased on a particular purchase order
Receive Date	Char		Optional	Date item was received from vendor.
Installation Date	Date		Optional	Date the item was installed. The system sets the value during EIN Installation processing.
Audit Date	Date		Optional	Date the item was physically inventoried last
EMD_MOD	Char	15	Optional	EMD Modification number.

611-EMD-001

Table 27.3.1-1. ILM-EIN Form Field Description (2 of 2)

Field Name	Data Type	Size	Entry	Description
Maintenance & Other Info.			Page	Contains the following fields about the EIN item: Maint Contract ID, Maint Exp Date, Maint Vendor, Warranty Exp Date, EMOSD ID, GFE Num, Comment, NASA Contract, Submitter, Create Date, and Last Modified By.
Maint Contract ID	Char	10	Optional	Identifier for the Maintenance Contract under which the item is covered.
Maint Exp Date	Date		Optional	Date the maintenance contract will expire. This field reflects the Expiration Date from the Maint Contract ID entered above.
GFE NUM	Char	8	Optional	Identifier assigned by the Government to an item of government furnished equipment.
EMOSD ID	Char	15	Optional	Identifier assigned by the EMOS Denver to an inventory item.
Comment	Char	120	Optional	Miscellaneous information specific to the item.
NASA Contract	Char	11	Optional, default NAS5-60000	Identifier designating the government contract used for this item.
Submitter	Char	30	System-supplied	The user whom created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user last modified the record.
Modify Date	Date		System-supplied	The last date the system was modified.
Components			Page	Page for displaying the components of a parent EIN. It displays the Component EIN, ECS Name, description, Serial No, Active Date, Inactive Date, Location, and Room.
Maintenance Contract			Page	Page displays attributes of the maintenance contract, such as Contract ID, Start Date, Expiration Date, Type of support, PO number, maintenance vendor, and vendor ID.
History			Page	Contains a listing of EIN transaction history for the EIN. This table displays the following fields describing the transactions: Trans Type, Date-Time, Operator ID, From Parent EIN, From ECS Name, From Location, From Room, To Parent EIN, To ECS Name, To Location, and To Room.

611-EMD-001

The following buttons are unique to this form:

- Add New Part – Activates the ILM-OEM Part form. This allows the operator to add new parts or to search for existing parts.
- EIN Transaction – brings up the ILM-EIN Transaction form. See section 27.3.3 for detail information about this form.

Table 27.3.1-2. Add New Inventory Item

Perform	Action	Expected Result
Navigate to the ILM-EIN Form	At the Remedy-ILM PC, ♦ **Start- > Programs -> Action Request System -> Remedy User** ♦ **File -> Open -> ILM-EIN -> New**	ILM-EIN form is displayed and ready for the user to add a new EIN record.
Defining new inventory item.	Fill in the necessary information: ♦ EIN: Enter the EIN number. If the item is an external hardware and has a silver sticker number, enter this number. If not, leave this field blank. ♦ Parent EIN: enter Parent EIN. Leave this field blank if the item is not a component to an EIN Structure. **Note:** Only structure hardware. ♦ Complete the Part information section. Note. The Part No field's label is **Bolded**. Therefore, you must enter a value into the Part No field. The Part No must exist in the ILM-OEM Part form. If not, the system will display an error message and will not except the value. If Part No not exist, click Add New Part button and add the part information. ♦ Complete the Location & Purchasing Info section. Note: The following fields are **bolded**: **Location, Item Status, Vendor ID, and PO Number**. Therefore, you must enter values in all of these fields in order to save the record. If not, the system will display an error message stating that all required fields must be completed and will not save the record. ♦ Complete the Maintenance & Other Info section. ♦ Click on the **Save** icon on the tool bar to save the record.	♦ When the Parent EIN is entered, the system will populate the following fields using the data derived from the Parent EIN record: ECS Name, Location, Building, Room, Vendor ID, PO Number, Item Status, Receive Date. ♦ If there is a value in the Parent EIN field, the system adds the EIN as a component to the Parent EIN Structure in the ILM-EIN Structure form using the current date as the active date. ♦ When the Part No is entered and the Part No exist in the ILM-OEM Part form, the system will populate the following fields: Description, MFR, Hw-Sw Code and Mod_Ver. ♦ If the EIN field is blank, the system will generate the next prefixed C number after the record is saved. The newly created EIN number is displayed in the Status bar on the bottom left of the form.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.3.1-3. Modifying EIN Record

Perform	Action	Expected Result
Navigate to the ILM-EIN Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-EIN -> Search** ♦ **Perform a Search for the records to be modified.**	♦ ILM-EIN form is displayed and ready for the user to add a new EIN record.
Modify ECS Name	♦ Find the EIN of interest. ♦ Enter the ECS Name. **Warning**: To maintain data consistency in the database, the system will not allow users to update the ECS Name of components. ♦ Click on the **Save** icon on the tool bar.	♦ The system updates the ECS Name for the EIN and all of its active components.
Modify the Audit Date	♦ Find the EIN of interest. ♦ Enter the new Audit Date. ♦ Click on the **Save** icon on the tool bar.	♦ The system updates the EIN's audit date as well as all the active C Number components audit date.

Note: To move to the next field use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

27.3.1.1   ILM-EIN Predefined Reports

Table 27.3.1.1-1 lists a set of special report designed specifically for the ILM-EIN form. These reports are built in Crystal Report Writer. Table 27.3.1.1-1 describes the purpose each of the reports serves. Table 27.3.1.1-2 provides instruction on how to run these reports. The sections that follow present the report layouts.
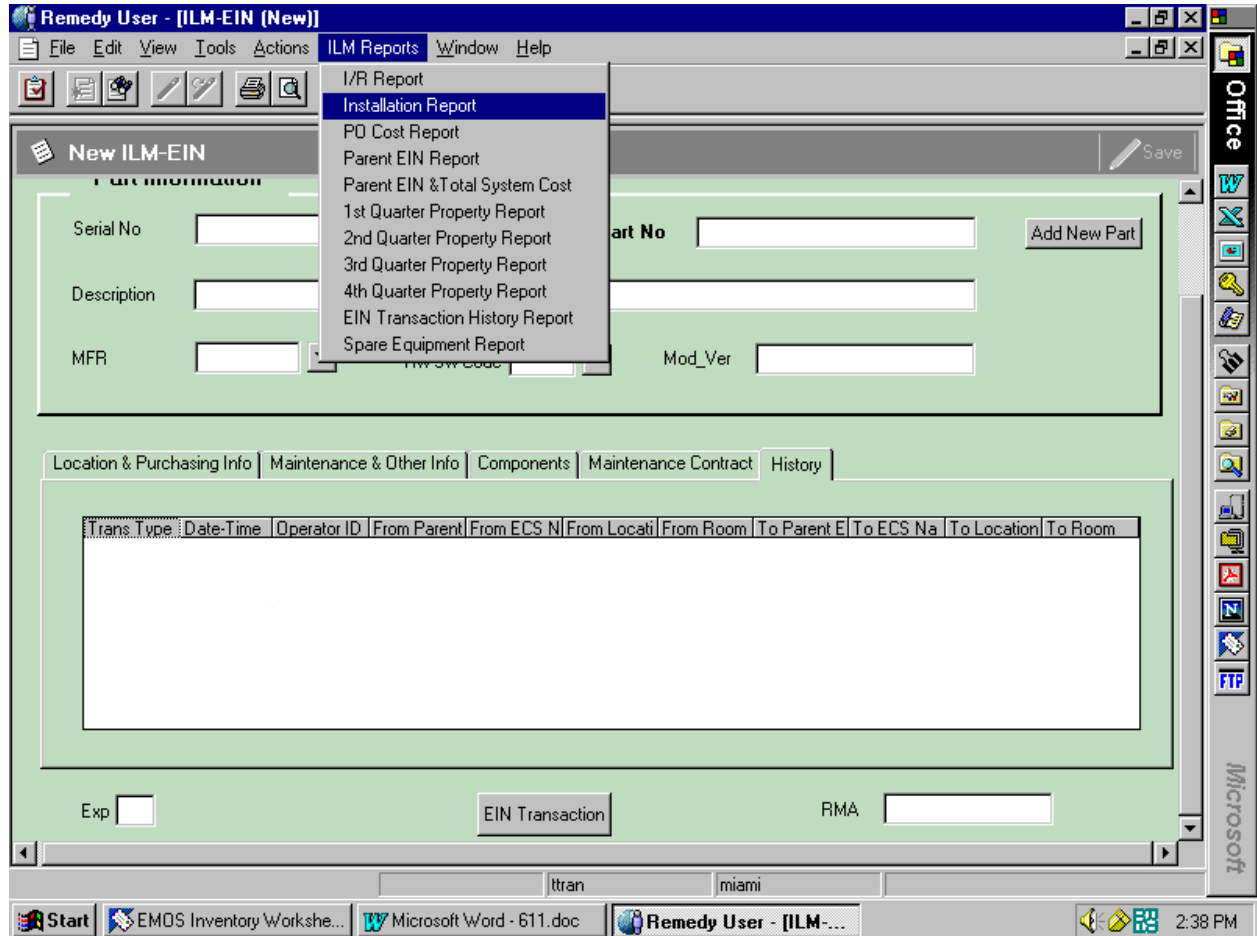
Figure 27.3.1.1-1. ILM-EIN Reports GUI

Table 27.3.1.1-1. ILM-EIN Pre-Defined Reports

Report Type	Report Description	Figure
Install/Receipt Report	A report describing an operator-specified EIN item together with all of its associated components order by EIN number.	27.3.1.1-6
Installation Report	A report describing an operator-specified EIN item together with its components having status "I" (for installed).	27.3.1.1-7
Purchase Order Cost Report	Provides a list of EINs and their cost associated with an operator-specified purchase order.	27.3.1.1-8
Parent EIN Report	Provides a listing of only Parent items.	27.3.1.1-9
Parent EIN and total System Cost Report	Provides a listing of only Parent items and the total system cost for each Parent.	27.3.1.1-10
Inventory Report	Provides an ASCII formated report identifying the inventory items by Parent EIN according to the operator-specified criteria.	27.3.1.1-11
Quarterly Property Management Report	Provides a list of contractor-acquired equipment items by quarter, sorted by Mfr and product description.	27.3.1.1-12
Cost – Selected ECS Managed Property	Provides the quantity and total cost of operator-selected EINs, grouped by type of inventory (Hardware, Software, Consumable, i.e.).	27.3.1.1-13
EIN Transaction History	A list of the transactions processed for operator-specified items during an operator-specified timeframe, sorted by EIN number and "from" location	27.3.1.1-14
Spare Equipment Report	Provides a list of spare equipment for a selected site or system-wide.	27.3.1.1-15

611-EMD-001

Table 27.3.1.1-2. Procedures to Generate ILM-EIN Predefined Reports (1 of 4)

Perform	Action	Expected Result
Navigate to the ILM-EIN Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-EIN -> Search**	♦ ILM-EIN form is displayed.
Run Install/Receipt Report	♦ Find the Parent EIN of Interest by doing a search in the **Parent EIN** field. ♦ When the records are displayed, **ILM Reports -> I/R Report** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Install/Receipt Report is displayed. Refer to Figure 27.3.1.1-6 for the report layout.
Run Installation Report	♦ Find the Parent EIN of Interest by doing a search in the **Parent EIN** field. ♦ When the records are displayed, **ILM Reports -> Installation Report** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Installation Report is displayed. Refer to Figure 27.3.1.1-7 for the report layout.
Run Purchase Order Cost Report	♦ Find the Purchase Order of Interest by doing a search in the **PO Number** field. ♦ When the records are displayed, **ILM Reports -> PO Cost Report** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Purchase Order Cost Report is displayed. Refer to Figure 27.3.1.1-8 for the report layout.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.3.1.1-2. Procedures to Generate ILM-EIN Predefined Reports (2 of 4)

Perform	Action	Expected Result
Run Parent EIN Report	♦ **ILM Reports -> Parent EIN Report** ♦ When the ILM-DIA Reports form is displayed (see Figure 27.3.1.1-2), select or enter a site name to run the report. Leaving the Site value blank will select all Parent EINs. ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Parent EIN report is displayed. Refer to Figure 27.3.1.1-9 for the report layout.
Run Parent EIN & Total System Cost Report	♦ **ILM Reports -> Parent EIN & Total System Cost Report** ♦ When the ILM-DIA Reports form is displayed (see Figure 27.3.1.1-2), select or enter a site name to run the report. Leaving the Site value blank will select all Parent EINs. ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Parent EIN & Total System Cost report is displayed. Refer to Figure 27.3.1.1-10 for the report layout.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

*Table 27.3.1.1-2. Procedures to Generate ILM-EIN Predefined Reports (3 of 4)*

Perform	Action	Expected Result

Run Inventory Report
Note: This report is not listed under ILM Report. It's located in the Remedy's Report Style listing.
	♦ Search for the records of interest to perform the report on.
	♦ When the records are displayed, make sure to un-select the highlighted record by pressing the **Ctrl** and single click the highlighted item simutaneously.
	♦ **Tools -> Reporting -> Inventory Rpt -> Report -> Export To -> File**
	♦ When the Report to File GUI (see Figure 27.3.1.1-3) is displayed.
	    Select the appropriate directory to save the report in
	    Enter a report file name
	    Save as Type = All File (*.*)
	Warning: The report will not work if the **Save as Type** is anything else other than All File.
	    Press the **Save** button
	♦ Bring up Excel
	♦ **File -> Open -> <Report Name>**
	♦ When the Text Import Wizard GUI (Figure 27.3.1.1-4) is displayed:
	    Delimiters –> Next
	    Select **Tab - > Next**
	    Change the EIN, Parent EIN, Part No, and Serial No fields to Text. This will keep all the leading zeros.
	    Click on the **Finish** button
		♦ The inventory report is displayed in Excel. User may make any necessary adjustment to this report. Please refer to MS Excel help for any question on how to use MS Excel. Refer to Figure 27.3.1.1-11 for the report layout.

Run (1st, 2nd, 3rd, or 4th) Quarterly Property Management Report	♦ **ILM Reports -> (**1st, 2nd, 3rd, or 4th **) Quarterly Property Report** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Quarterly Property Management report is displayed. Refer to Figure 27.3.1.1-12 for the report layout.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.3.1.1-2. Procedures to Generate ILM-EIN Predefined Reports (4 of 4)

Perform	Action	Expected Result
Run Cost-Selected ECS Managed Report **Note:** This report is not listed in the ILM Reports listing. It's locates in the Remedy's Report Style.	♦ Search for the records of interest to perform the report. ♦ When the records are displayed, make sure to un-select the highlighted record by pressing the **Ctrl** and single click the highlighted item simultaneously. ♦ **Tools -> Reporting -> Cost – ECS Managed Property -> Report -> Preview** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Cost-Selected ECS Managed Report is displayed. See Figure 27.3.1.1-13 for report layout.
Run EIN Transaction History Report	♦ Find the EIN of Interest by entering the EIN in the **EIN** field and performing a search. ♦ When the records are displayed, **ILM Reports -> EIN Transaction History Report** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The EIN Transaction History report is displayed. See Figure 27.3.1.1-14 for report layout.
Run Spare Equipment Report	♦ **ILM Reports -> Spare Equipment Report** ♦ When the ILM-DIA Reports form is displayed (see Figure 27.3.1.1-2), select or enter a site name to run the report. Leaving the Site value blank will select all the spare equipment in the database. ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Spare Equipment report is displayed. See Figure 27.3.1.1-15 for report layout.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.
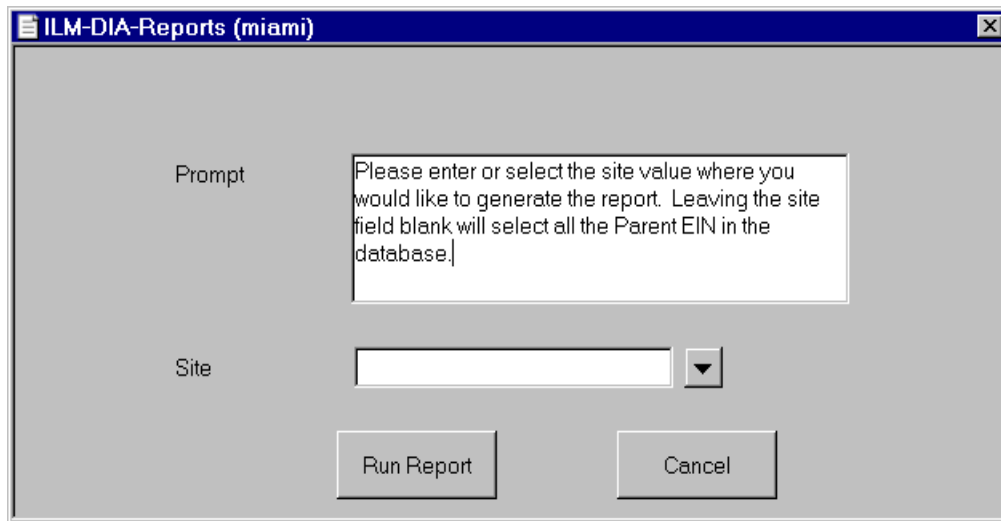
Figure 27.3.1.1-2. ILM-DIA Reports GUI
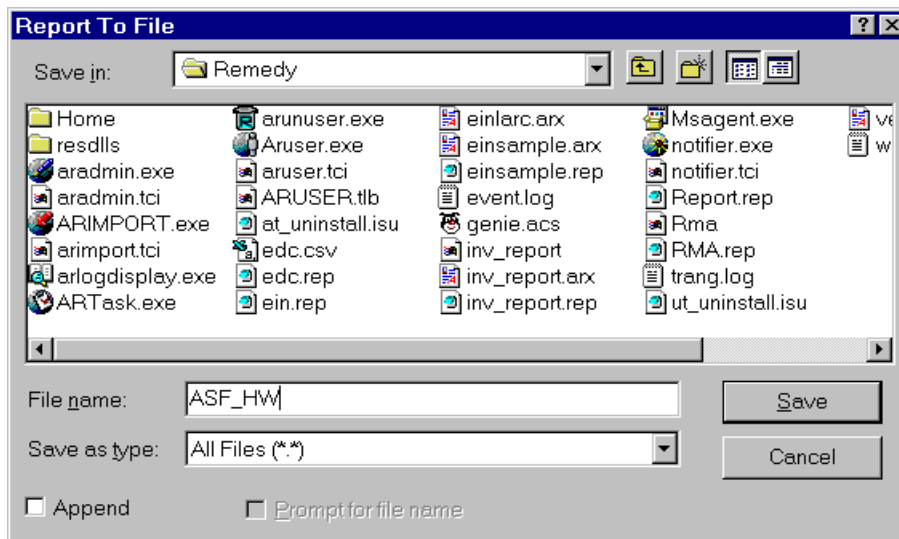
Figure 27.3.1.1-3. Report To File GUI
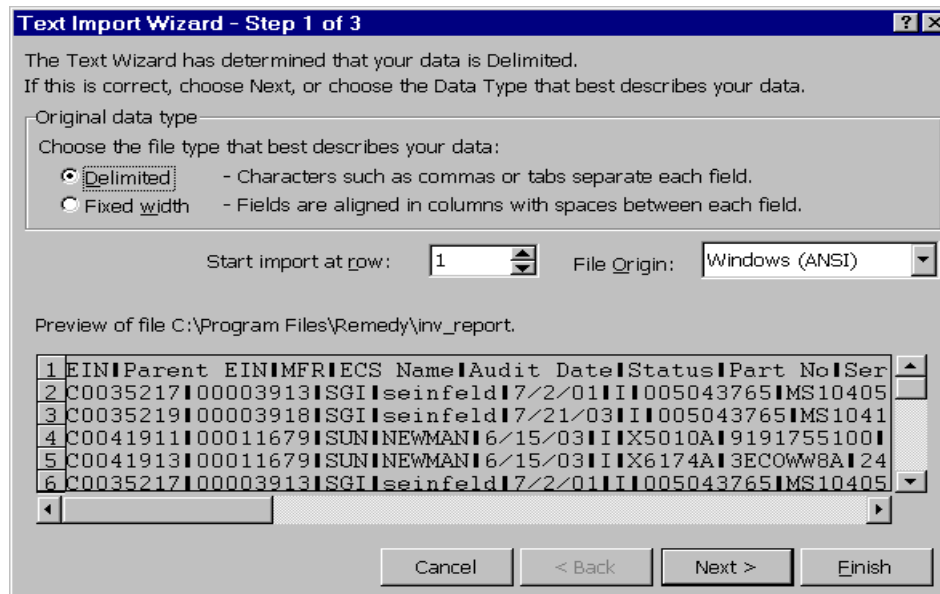
Figure 27.3.1.1-4. Text Import Wizard GUI
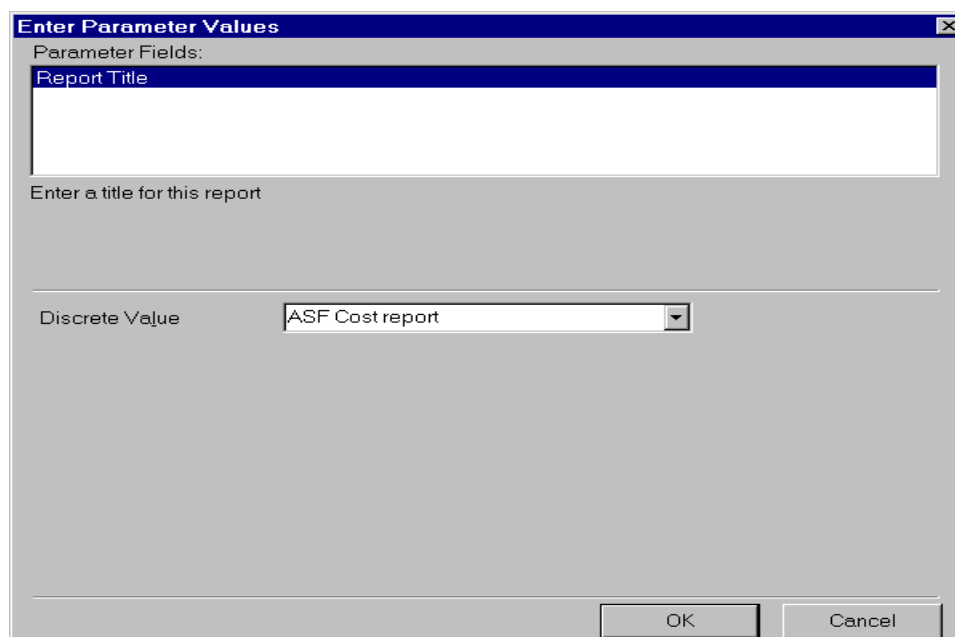
Figure 27.3.1.1-5. Enter Parameter Values GUI

Acrobat Reader - [~IL203.pdf]

File   Edit   Document   View   Window   Help

Figure 27.3.1.1-6. Install/Receipt Report GUI

Figure 27.3.1.1-7. Installation Report GUI

Acrobat Reader - [~IL263.pdf]

File   Edit   Document   View   Window   Help

DATE:6/30/03   1:28:53PM   PURCHASE ORDER COST REPORT   Page 1 of 1
                                                        VENDOR~ SGI
                                                  PO Number ~ CCL0014185

EIN	PART NUM	DESCRIPTION	SERIAL NO	COST
00020620	WF-600V10-2073	FUEL V10 GRAPHINS 600MHZ	0800691051C4	$14,054.00
00020621	M-543	MOUSE		$0.00
00020622	KBB-US	KEYBOARD		$0.00
C0050414	PCIX-GIGENET-C	1 PORT COPPER GB	HYUR126954	$511.00
C0050415	PCIX-GIGENET-OR-SU	1 PORT OPT ETHERNET CARD	HYTR126839	$1,056.00
C0050418	P10-CDR48INT	INTERNAL CD-ROM 40X		$594.00
C0050419	P10-73G10K-INT	10000RPM INT 3.5 73GB		$1,584.00

                                                                                        $17,799.00

120%   1 of 1   11 x 8.5 in

Figure 27.3.1.1-8. Purchase Order Cost Report GUI

Figure 27.3.1.1-9. Parent EIN Report GUI

Figure 27.3.1.1-10. Parent EIN & Total System Cost Report GUI

Parent EIN	Part EIN	Mfr	ECS Name	Audit Date	Stat	Part Num	Serial Num	Unit Cost	Product Description	Location	Bldg	Room	Code	PO Num	Date Rec'd	Vendor
00001838	00001838	SUN	n0dms04	2/6/02	I	A12-UBA1-1E-064AB	645F0AA4	$8,797.00	Ultra 1 System	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	00001891	SUN	n0dms04	2/6/02	I	X5203A	645G0830	$1,035.00	UniPak - 4.2 GB 5400 RPM FW SCSI-2	NSIDC	NSIDC	209	H	CCW0005354	12/18/96	SUN
00001838	00003491	SUN	n0dms04	2/6/02	I	X267A	9843KN4545	$5,000.00	Color Monitor - 20 IN	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	00006793	SUN	n0dms04	2/6/02	I	NE SUN1	LZB64001097	$0.00	Mouse - 3 Button Track Ball	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	00006794	SUN	n0dms04	2/6/02	I	320-1233-02	9626371319	$0.00	Keyboard	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	C0009132	SUN	n0dms04	2/6/02	I	X1025A	12603	$1,500.00	FDDI SINGLE ATTACH SBUS CARD	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	C0009133	SUN	n0dms04	2/6/02	I	X132P	501282278476877	$0.00	Memory - 32MB RAM Expansion (1x32	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	C0009134	SUN	n0dms04	2/6/02	I	X132P	501262278476942	$0.00	Memory - 32MB RAM Expansion (1x32	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	C0009135	SUN	n0dms04	2/6/02	I	X3500A		$0.00	Country Kit	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	C0009136	SUN	n0dms04	2/6/02	I	X6001A	9625201083	$90.00	Floppy Drive - 3.5 IN Disk Drive-1	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	C0009137	SUN	n0dms04	2/6/02	I	X6153A	9647723061	$240.00	CD ROM - Internal SUNCD 4	NSIDC	NSIDC	209	H	CCW0005354	12/2/96	SUN
00001838	C0009839	SUN	n0dms04	2/6/02	I	370-2286-01	9643604099	$0.00	Hard Drive - Internal for X5204A	NSIDC	NSIDC	209	H	CCW0005354	12/18/96	SUN
00001838	C0045377	SUN	n0dms04	6/25/02	I	X5237A	0145KP6EMS	$479.20	ULTRA SCSI DISK DRIVE, 18GB INTERN	NSIDC	NSIDC	209	H	CCD0002848	11/27/01	SUN
00001838	C0149507	SUN	n0dms04	1/27/99	I	SOLD-C	11748	$45.00	Solaris Media for Servers	NSIDC	NSIDC	209	S	CCW0005354	12/2/96	SUN
00001839	00001839	SUN	n0mos20	2/6/02	I	A12-UBA1-1E-064AB	645F0B2C	$8,797.00	Ultra 1 System	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	00001890	SUN	n0mos20	2/6/02	I	X5203A	645G0868	$1,035.00	UniPak - 4.2 GB 5400 RPM FW SCSI-2	NSIDC	NSIDC	252	H	CCW0005354	12/18/96	SUN
00001839	00003494	SUN	n0mos20	2/6/02	I	X267A	9647GI3704	$5,000.00	Color Monitor - 20 IN	NSIDC	NSIDC	252	H	CCW0005354	4/3/00	SUN
00001839	00006771	SUN	n0mos20	2/6/02	I	320-1233-02	9626371388	$0.00	Keyboard	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	00006772	SUN	n0mos20	2/6/02	I	NE SUN1	LZB64001084	$0.00	Mouse - 3 Button Track Ball	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	C0009138	SUN	n0mos20	2/6/02	I	370-2040-03	9643547711	$0.00	Hard Drive - 2.1 GB Internal	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	C0009139	SUN	n0mos20	2/6/02	I	X1025A	11748	$1,500.00	FDDI SINGLE ATTACH SBUS CARD	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	C0009140	SUN	n0mos20	2/6/02	I	X132P	501262279495644	$0.00	Memory - 32MB RAM Expansion (1x32	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	C0009141	SUN	n0mos20	2/6/02	I	X132P	501262279495714	$0.00	Memory - 32MB RAM Expansion (1x32	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	C0009142	SUN	n0mos20	2/6/02	I	X3500A		$0.00	Country Kit	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	C0009143	SUN	n0mos20	2/6/02	I	X6001A	9625201087	$90.00	Floppy Drive - 3.5 IN Disk Drive-1	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	C0009144	SUN	n0mos20	2/6/02	I	X6153A	6Y50C01029	$240.00	CD ROM - Internal SUNCD 4	NSIDC	NSIDC	252	H	CCW0005354	12/2/96	SUN
00001839	C0009838	SUN	n0mos20	2/6/02	I	370-2286-01	9643604145	$0.00	Hard Drive - Internal for X5204A	NSIDC	NSIDC	252	H	CCW0005354	12/18/96	SUN
00001839	C0038808	SUN	n0mos20	2/6/02	I	501-2961	017403	$0.00	System Board	NSIDC	NSIDC	252	H	CCW0005354	4/16/01	SUN
00001839	C0149508	SUN	n0mos20	1/27/99	I	SOLD-C		$45.00	Solaris Media for Servers	NSIDC	NSIDC	252	S	CCW0005354	12/2/96	SUN

Figure 27.3.1.1-11. Inventory Report GUI

Acrobat Reader - [~IL244.pdf]

File Edit Document View Window Help

6/30/03 1:27:06PM
ECS Quarterly Property Management Report
Contractor Acquired Equipmemt
Contract number NAS5-60000
Period: 4/1/03 to 6/30/03
Page 1 of 1

EIN	MFR	DESCRIPTION	SERIAL NO	LOCATION	COST
C0001851	ATI	10 BASE T TRANSCEIVER		EDFARC	$39.00
00000999	ATI	10 BASE T TRANSCEIVER		EDF	$0.00
00000998	ATI	10 BASE T TRANSCEIVER		EDF	$0.00
C0051060	STK	LC/SC 50M/120U FIBER CABLE		EDF	$126.00
C0051061	STK	FC 50/ 125 SC-SC DUP 50 MTR		EDF	$206.00
C0051094	SGI	25M OPT CABLE W/LC/LC CONNETORS		EDF	$340.56
C0051128	SUN	CABLE ULTRA SCSI-3		EDF	$152.00
C0051136	MIT	MITSUI DVD-R		EDF	$3,175.00
C0051137	MIT	JEWEL CASE ON SIDE		EDF	$100.00
C0051154	MBS	J-CARD FRONT INSERT 80#		EDF	$35.00
00030000	SUN	SPARCSTATION 20-71 SX	LKJSDFLJ	EDF	$4,522.00
C5000046	SUN	CD ROM - INTERNAL SUNCD 2 PLUS	LKEIOLKJASD	EDF	$555.00
C5000047	SUN	HARD DRIVE - 8.4GB F/W SCSI-2 DESKTOP	ILEJW90ELKJDA	EDF	$555.00
C5000052	SGI	18GB 10K EXTERNAL HD	1234567890123456789	EDF	$0.00
C5000053	SGI	18GB 10K EXTERNAL HD	1234567890123456789	EDF	$0.00
C5000054	SUN	INTERNAL 18.2GB ULTRA SCSI DRIVE	KLJASDJFKLSD56432132	EDF	$0.00
C5000058	SUN	INTERNAL 18.2GB ULTRA SCSI DRIVE	6523ASD56FDS321SD	EDF	$0.00
C5000068	IBM	TESTING MWO LI ADD NEW PART	KLJKJDAIODKL	EDF	$0.00

The cost of these contractor-acquired equipment for the quarter ending 6/30/03 is: $9,805.56

120%   1 of 1   11 x 8.5 in

Figure 27.3.1.1-12. Quarterly Property Management Report GUI

Acrobat Reader - [~Co274.pdf]

File   Edit   Document   View   Window   Help

DATE: 6/30/03   COST - SELECTED ECS MANAGED PROPERTY   PAGE:1

You can specify the report name here.

	QTY	COST
Items coded C:(Consumable)	8	$5,301.00
Items coded H:(Hardware)	12	$56,453.26
Total:	20	$61,754.26

156%   1 of 1   8.5 x 11 in

Figure 27.3.1.1-13. Cost – Selected ECS Managed Report GUI

Acrobat Reader – [~IL280.pdf]
File Edit Document View Window Help

Date: 6/30/03 1:32:07PM **EIN Transaction History Report** Page 1 of 1

EIN: C0001851
ECS_Name: IN STOCK
Description: 10 BASE T TRANS
Part_No: AT210TS
Serial_No:
MFR: ATI
Location: EDF
Room: 1073
Status: R

Trans Type	Date	Operator ID	From Location	To Location	From Parent	To Parent
RTS	6/4/03	ilmuser	EDF	EDF	00000127	
MFV	6/4/03	ilmuser	EDF	EDFARC	00001881	
REL	6/25/03	ttran	EDFARC		00000127	
REL	6/26/03	ttran	EDFARC		00000127	

156%   1 of 1   8.5 x 11 in

Figure 27.3.1.1-14. EIN Transaction History Report
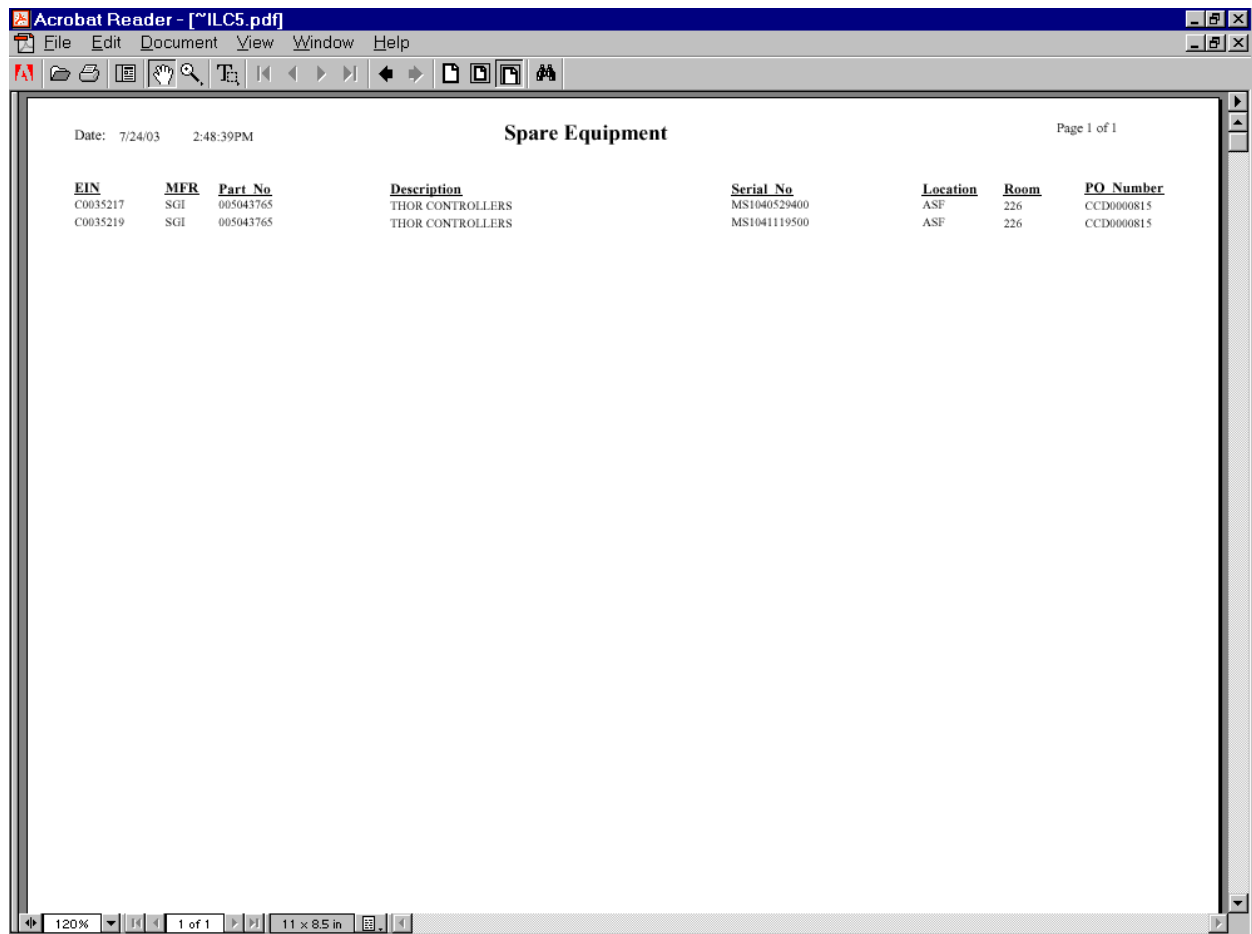
Figure 27.3.1.1-15. Spare Equipment Report GUI

27.3.2 ILM-EIN Structure GUI

The ILM-EIN Structure form (Figures 27.3.2-1 and 27.3.2-2) is designed to allow an Administrator to repair EIN structure records. Other ILM groups may view EIN Structure via the ILM-EIN form discussed in the previous section. To make changes to EIN Structures use the ILM-EIN Transaction form discussed in the next section.

Figure 27.3.2-1. ILM-EIN Structure GUI (1 of 2)

Figure 27.3.2-2. ILM-EIN Structure GUI (2 of 2)

Table 27.3.2-1 contains description of the ILM-EIN Structure form fields.

Table 27.3.2-1. ILM-EIN Structure Field Descriptions

Field Name	Data Type	Size	Entry	Description
Parent EIN	Char	20	Required	EIN for the parent item in an EIN structure.
ECS Name	Char	30	System-supplied	Name of the machine with which the item is associated.
System Serial No	Char	30	System-supplied	Serial number of the item.
Part No	Char	34	System-supplied	Manufacturer's part number.
Description	Char	60	System-supplied	Manufacturer's or vendor's description for the item.
MFR	Char	6	System-supplied	Code for the manufacturer of the item.
Location	Char	6	System-supplied	Identifier that designates an inventory location.
Building	Char	6	System-supplied	The building where the item can be found.
Room	Char	15	System-supplied	The room where the item can be found.
Component EIN	Char	20	Required	Identifier for an EIN controlled inventory item.
Active Date	Date		Required	Date the item was added to the parent structure.
Inactive Date	Date		Optional	Date the component is no longer assigned to the Parent EIN.
Process	Char	1	Optional	Identifier for Component EIN to be processed by EIN transactions.
Components	Table field		System-supplied	Field for displaying the components of a parent EIN. This table contains the following fields describing the Component EIN: Component EIN, ECS Name, Description, Serial No, Active Date, Inactive Date, Location, and Room.

27.3.3   ILM-EIN Transaction GUI

The ILM-EIN Transaction form (Figures 27.3.3-1 through 27.3.3-3) lets the operator perform the following EIN transactions for inventory items: Installation, Relocation, Return to Stock, Movement, Shipment, and Archive. The operator may select the type of transaction from the drop down list on the Transaction Type field as displayed below. Depending on the type of transaction the operator selects, Remedy will perform inventory updates accordingly. The Operator can specify components to be processed by pressing the "Select Components to Process" button. Remedy then transfer the operator to the ILM-Process Component form to complete the transaction. Table 27.3.3-1 describes the field definitions for this form. Table 27.3.3-2 presents detail instructions on how to perform these transactions and how each of them affects the property records and their EIN structures.

Figure 27.3.3-1. ILM-EIN Transaction GUI (Install/Move/Ship/RTS)

611-EMD-001

Figure 27.3.3-2. ILM-EIN Transaction GUI (Relocation)

Figure 27.3.3-3. ILM-EIN Transaction GUI (Archive)

Table 27.3.3-1 describes the ILM-EIN transaction form field definitions.

Table 27.3.3-1. ILM-EIN Transaction Form Field Descriptions (1 of 2)

Field Name	Data Type	Size	Entry	Description
Parent EIN	Char	20	Required	EIN for the parent item in an EIN structure.
Transaction Type	Char	15	Required	Type of transaction performs on the Parent EIN such as Installation, relocation, movement, shipment, Return To Stock, and archive.
Effective Date	Date		Optional	The date the transaction is in effect.
ECS Name	Char	30	System-supplied	Name of the machine with which the item is associated.
System Serial No	Char	30	System-supplied	Serial number of the Parent EIN.
Part No	Char	34	System-supplied	Manufacturer's or vendor's part number.
MFR	Char	6	System-supplied	Code for the manufacturer of the item.
Description	Char	60	System-supplied	Manufacturer's or vendor's description for the item.
Location	Char	6	System-supplied	Identifier that designates an inventory location.
Room	Char	15	System-supplied	The room where the item can be found.
New ECS Name	Char	30	Optional	New ECS Name for the Parent EIN.
New Location	Char	6	Optional	New Location where the item will be.
New Building	Char	6	Optional	New Building where the item will be.
New Room	Char	15	Optional	New room where the item will be.
Return (P)arent-(C)omponent	Char	1	Optional, P or C	Identify whether the operator will return Parent and all of the components to stock or return some of the components to stock.
Ship (P)arent-(C)omponent	Char	1	Optional, P or C	Identify whether the operator will ship Parent and all of the components or ship some of the components to a new location.
New Parent EIN	Char	20	Optional	New Parent EIN to which the item(s) will be associated with.
Archive (P)arent-(C)omponent	Char	1	Optional, P or C	Identify whether the operator will archive the Parent as well as all the active components or archive a subset of components.
Type of Archive	Char	6	Optional, X, TV,G, RG	Define the type of archive, ie: Return to Vendor – X; Trade in to vendor – TV; Transferred to government – G; Government Relieved Accountability – RG; and Excess sold to vendor = S

Table 27.3.3-1. ILM-EIN Transaction Form Field Descriptions (2 of 2)

Field Name	Data Type	Size	Entry	Description
Install/Move/Ship/ RTS	Page		N/A	This page contains the following fields to perform the EIN Installation, Movement, Shipment, and Return to Stock: Return (P)arent-(C)omponent, Ship (P)arent-(C)omponent, New ECS Name, New Location, New Building, and New Room.
Relocate	Page		N/A	This page contains the New Parent EIN field for user to perform EIN relocation.
Archive	Page		N/A	This page contains the following fields to perform EIN archive: Archive (P)arent-(C)omponent and Type of Archive.

♦ Pressing the Execute Transaction will process the transaction and updates inventory items according to the type of transaction the operator selected.

♦ Pressing the "Select Components to Process" button will bring up the ILM-Process Component form. This button will be visible only when the transaction is associated with components.

Table 27.3.3-2. Procedures to Perform EIN Transactions (1 of 8)

Perform	Action	Expected Result
Navigate to the ILM-EIN Transaction	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-EIN Transaction-> New**	ILM-EIN Transaction form is displayed.

Installing a Parent and all of it component – Installation is designed for updating property records to reflect installation of inventory items.	♦ Enter **Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Installation** from the list of Transaction Types
	♦ Enter **New ECS Name, New Location, New Building, and New room**
	♦ Press the **Execute Transaction** button on the bottom of the form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the Parent EIN and all of its active components:
		♦ ECS Name = New ECS Name
		♦ Location = New Location
		♦ Building = New Building
		♦ Room = New Room
		♦ Item Status = I (Install)
		♦ Installation and Audit Date = Effective Date
		♦ Writes an "INS" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, and ECS Name and Location values changes.

Moving a Parent and all of its components. EIN Movement is designed for updating property records when a Parent EIN and all of its children are being moved from one room within a site to another.
Note: Use Shipment for inter-site movement.
	♦ Enter **Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Movement** from the list of Transaction Types
	♦ Enter **New ECS Name, New Building, and New room**
	♦ Press the **Execute Transaction** button on the bottom of the form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the Parent EIN and all of its active components:
		♦ ECS Name = New ECS Name
		♦ Building = New Building
		♦ Room = New Room
		♦ Item Status = I (Install)
		♦ Installation and Audit Date = Effective Date
		♦ Writes an "MV" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, and ECS Name and Location values changes.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.3.3-2. Procedures to Perform EIN Transactions (2 of 8)

Perform	Action	Expected Result

Shipping a Parent and all of its components	♦ Enter **Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Shipment** from the list of Transaction Types
	♦ Select **P** from the Ship (P)arent-(C)omponent field.
	♦ Enter **New ECS Name, New Location, New Building, and New room**
	♦ Press the **Execute Transaction** button on the bottom of the form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the Parent EIN and all of its active components:
		♦ ECS Name = New ECS Name
		♦ Location = New Location
		♦ Building = New Building
		♦ Room = New Room
		♦ Item Status = I (Install)
		♦ Installation and Audit Date = Effective Date
		♦ Writes an "SHP" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, and ECS Name and Location values changes.

Note: To move to the next field, use either the TAB key or the Mouse.   The Enter key will not move the cursor to the next field.

Table 27.3.3-2. Procedures to Perform EIN Transactions (3 of 8)

Perform	Action	Expected Result

Shipping Components only	♦ Enter **Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Shipment** from the list of Transaction Types
	♦ Select **C** from the Ship (P)arent-(C)omponent field.
	♦ Enter **New ECS Name, New Location, New Building, and New room**
	♦ Press the "**Select Components To Process**" button
	♦ When the ILM-Join Process Component form (Figure 27.3.3.1-1) is displayed, find the component to be processed listed in the result list.
	♦ Select **Y** from the Process field.
	♦ Press the **Save** icon.
	♦ When finished specifying components to process, close the ILM-Join Process Component form.
	♦ Press the **Execute Transaction** button on the bottom of the ILM-EIN Transaction form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the selected components:
		♦ Parent EIN = Blank
		♦ ECS Name = New ECS Name
		♦ Location = New Location
		♦ Building = New Building
		♦ Room = New Room
		♦ Item Status = R (Receive)
		♦ Installation date = Blank
		♦ Audit Date = Effective Date
		♦ Writes an "SHP" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, and ECS Name, Parent EIN, and Location values changes.
		♦ Inactivate the selected components from the EIN Structure using the Effective Date as the Inactive date.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.3.3-2. Procedures to Perform EIN Transactions (4 of 8)

Perform	Action	Expected Result

Moving Parent and Components to stock	♦ Enter **Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Return To Stock** from the list of Transaction Types
	♦ Select **P** from the Return (P)arent-(C)omponent field.
	♦ Enter **New ECS Name, New Location, New Building, and New room**
	♦ Press the **Execute Transaction** button on the bottom of the form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the Parent EIN and all of its active components:
		♦ ECS Name = IN STOCK
		♦ Location = New Location
		♦ Building = New Building
		♦ Room = New Room
		♦ Item Status = R (Receive)
		♦ Installation Date = BLANK
		♦ Audit Date = Effective Date
		♦ Writes an "RTS" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, ECS Name, and Location values changes.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.3.3-2. Procedures to Perform EIN Transactions (5 of 8)

Perform	Action	Expected Result

Moving components to stock	♦ Enter **Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Return To Stock** from the list of Transaction Types
	♦ Select **C** from the Return (P)arent-(C)omponent field.
	♦ Enter **New Location, New Building, and New room**
	♦ Press the "**Select Components To Process**" button
	♦ When the ILM-Join Process Component form (Figure 27.3.3.1-1) is displayed, find the component to be processed listed in the result list.
	♦ Select **Y** from the Process field.
	♦ Press the **Save** icon.
	♦ When finished specifying components to process, close the ILM-Join Process Component form.
	♦ Press the **Execute Transaction** button on the bottom of the ILM-EIN Transaction form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the selected components:
		♦ Parent EIN = Blank
		♦ ECS Name = IN STOCK
		♦ Location = New Location
		♦ Building = New Building
		♦ Room = New Room
		♦ Item Status = R (Receive)
		♦ Installation Date = BLANK
		♦ Audit Date = Effective Date
		♦ Writes an "RTS" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, and ECS Name, Parent EIN, and Location values changes.
		♦ Inactivate the selected components from the EIN Structure using the Effective Date as the Inactive date.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

611-EMD-001

Table 27.3.3-2. Procedures to Perform EIN Transactions (6 of 8)

Perform	Action	Expected Result

Relocating components to a new EIN Structure	♦ Enter **Current Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Relocation** from the list of Transaction Types
	♦ Enter the **new Parent EIN** into the New Parent EIN field.
	♦ Press the "**Select Components To Process**" button
	♦ When the ILM-Join Process Component form (Figure 27.3.3.1-1) is displayed, find the component to be processed listed in the result list.
	♦ Select **Y** from the Process field.
	♦ Press the **Save** icon.
	♦ When finished specifying components to process, close the ILM-Join Process Component form.
	♦ Press the **Execute Transaction** button on the bottom of the ILM-EIN Transaction form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the selected components:
		♦ Parent EIN = New Parent EIN
		♦ ECS Name = New Parent's ECS Name
		♦ Location = New Parent's Location
		♦ Building = New Parent's Building
		♦ Room = New Parent's Room
		♦ Item Status = I (Install)
		♦ Installation and Audit Date = Effective Date
		♦ Writes an "REL" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, and ECS Name, Parent EIN and Location values changes.
		♦ Inactivate the selected components from the Old EIN Structure using the Effective Date as the Inactive date.
		♦ Add the selected components to the New EIN Structure using the Effective Date as the active date.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.3.3-2. Procedures to Perform EIN Transactions (7 of 8)

Perform	Action	Expected Result

Archivng a Parent and all of its components	♦ Enter **Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Archive** from the list of Transaction Types
	♦ Select **P** from the Archive (P)arent-(C)omponent field.
	♦ Select the type of archive from the list of archive types:
	    Return to Vendor (X)
	    Trade in to Vendor (TV)
	    Excess sold to vendor (S)
	    Transferred to Government (G)
	    Government Relieved from accountability (RG)
	♦ Press the **Execute Transaction** button on the bottom of the form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the Parent EIN and all of its active components:
		♦ ECS Name = ARCHIVE
		♦ Location = EDFARC
		♦ Building = BLANK
		♦ Room = BLANK
		♦ Item Status = Depends on type of archive selected.
		♦ Installation Date = BLANK
		♦ Audit Date = Effective Date
		♦ Writes an "ARC" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, and ECS Name and Location values changes.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

27-55 611-EMD-001

Table 27.3.3-2. Procedures to Perform EIN Transactions (8 of 8)

Perform	Action	Expected Result

Archiving components only	♦ Enter **Parent EIN** into the Parent EIN field
	♦ Enter the **Effective Date**
	♦ Select **Archive** from the list of Transaction Types
	♦ Select **C** from the Archvie (P)arent-(C)omponent field.
	♦ Select the type of archive from the list of archive types:
	    Return to Vendor (X)
	    Trade in to Vendor (TV)
	    Excess sold to vendor (S)
	    Transferred to Government (G)
	    Government Relieved from accountability (RG)
	♦ Press the "**Select Components To Process**" button
	♦ When the ILM-Join Process Component form (Figure 27.3.3.1-1) is displayed, find the component to be processed listed in the result list.
	♦ Select **Y** from the Process field.
	♦ Press the **Save** icon.
	♦ When finished specifying components to process, close the ILM-Join Process Component form.
	♦ Press the **Execute Transaction** button on the bottom of the ILM-EIN Transaction form.
	♦ The ILM-TRS-Dialog form (Figure 27.3.3-4) displayed, asking the user "Would you like to perform another transaction?" and provides the following options:
	    Same Parent EIN
	    Another Parent EIN
	    Exit EIN Transaction
	♦ Press "Another Parent EIN" button to perform another transaction.
		Remedy performs the following updates for the selected components:
		♦ Parent EIN= Blank
		♦ ECS Name = ARCHIVE
		♦ Location = EDFARC
		♦ Building = BLANK
		♦ Room = BLANK
		♦ Item Status = Depends on type of archive selected.
		♦ Installation Date = BLANK
		♦ Audit Date = Effective Date
		♦ Writes an "ARC" record in the transaction log to capture details of the event. This includes date/time of the event, operator initiating it, and ECS Name, Parent EIN and Location values changes.
		♦ Inactivate the selected components from the EIN Structure using the Effective Date as the Inactive date.
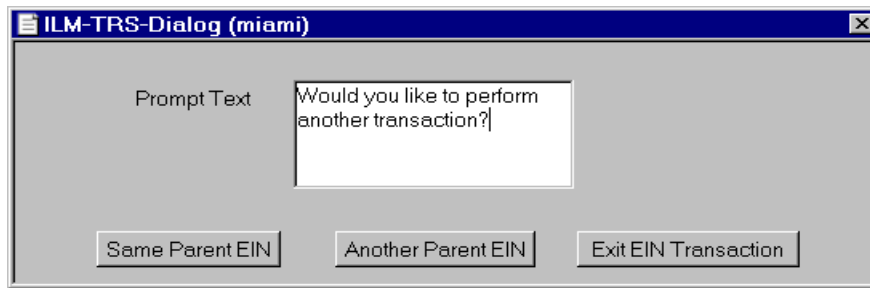
Figure 27.3.3-4. ILM-TRS Dialog GUI

27.3.3.1 ILM-Process Component GUI

The ILM-Process Component form (Figure 27.3.3.1-1) displays all the active components for the Parent EIN entered on the ILM-EIN Transaction form. The User may specify components to be processed by entering a "Y" in the Process field. This form can be accessed through the "Select Components To Process" button on the buttom of the ILM-EIN Transaction form. This button is only visible when the transaction is being performed on components. For example, when the user selects to return components (Return (P)arent-(C)omponent) to stock, or relocate components to a new EIN Structure, or archive components.

611-EMD-001

Figure 27.3.3.1-1. ILM-Process Component GUI

Table 27.3.3.1-1 describes the field definitions for the ILM-Process Component Form.

Table 27.3.3.1-1. ILM-Component to Process Field Descriptions

Field Name	Data Type	Size	Entry	Description
Parent EIN	Char	20	System-supplied	EIN for the parent item in an EIN structure.
ECS Name	Char	30	System-supplied	Name of the machine with which the item is associated.
Component EIN	Char	20	System-supplied	EIN for the Component item in an EIN structure.
Process	Char	1	Optional	Indicates whether or not a component is to be processed. Y = Yes, N = No.
Part No	Char	34	System-supplied	Manufacturer's or vendor's part number.
MFR	Char	6	System-supplied	Code for the manufacturer of the item.
Description	Char	60	System-supplied	Manufacturer's or vendor's description for the item.
Serial No	Char	30	System-supplied	Serial number of the item.
Item Status	Char	1	System-supplied	Code that designates the status of the item.
Location	Char	6	System-supplied	Identifier that designates an inventory location.
Room	Char	15	System-supplied	The room where the item can be found.
Active Date	Date		System-supplied	Date the item was added to the parent structure.
Inactive Date	Date		System-supplied	Date the component is no longer assigned to the EIN Structure.

611-EMD-001

27.3.4 ILM-Transaction Log

ILM-Transaction Log form (Figures 27.3.4-1 and 27.3.4-2) is designed for viewing/browsing all the EIN transactions performed on property records. Remedy logs the type of transaction, date/time, operator initiating the transaction, ECS name, Parent EIN, and location changes. In addition, this form also shows property record changes due to maintenance actions performed on inventory items (refer to section 27.4).

Figure 27.3.4-1. ILM-Transaction Log GUI (1 of 2)

Figure 27.3.4-2. ILM-Transaction Log GUI (2 of 2)

Table 27.3.4-1 describes the fields on the ILM-Transaction form.

Table 27.3.4-1. ILM-Transactions Field Descriptions (1 of 2)

Field Name	Data Type	Size	Entry	Description
Transaction No	Numeric	10	System-supplied	A system generated number that uniquely identifies the transaction.
Trans Type	Char	5	System-supplied	The type of transaction operators perform on an inventory item, including: INS - Install, MVE – Move, REL - Relocate, ARC – Archive, SHP – Ship, RTS – Return to Stock, MFS – Failed and Returned to Stock, MFV – Failed and returned to the vendor, MNS – New and came from stock, MNV – New and came from vendor, MRR – Relocate to a new Parent via the MWO, and MRS – Relocate to stock via the MWO.
Date-Time	Date		System-supplied	Date and time the transaction occurred.
Operator ID	Char	10	System-supplied	The operator id who performed the transaction.
EIN	Char	20	System-supplied	The EIN number that the transaction performed on.
ECS Name	Char	30	System-supplied	Name of the machine with which the item is associated.
Serial No	Char	30	System-supplied	Serial number of the item.
Part No	Char	30	System-supplied	Manufacturer's or vendor's part number.
Description	Char	60	System-supplied	Manufacturer's or vendor's description of the item.
MFR	Char	6	System-supplied	Code for the manufacturer of the item.
Item Status	Char	2	System-supplied	Code that designates the status of the item. The following values are set when processing transactions: R = Received; SP = Spare Equipment; I = Installed; X = Returned to vendor; G = Transfer to Gov; TV = Trade in to Vendor; S = Excess sold to vendor; RG = Relieved from accountability
Location	Char	6	System-supplied	Identifier designates an inventory location.
Building	Char	6	System-supplied	Identifier for the building where the item can be found.
Room	Char	15	System-supplied	Identifier for the room where the item can be found.
From Parent EIN	Char	20	System-supplied	The parent EIN where the EIN originated.
To Parent EIN	Char	20	System-supplied	The new parent EIN where the EIN is locating.

611-EMD-001

Table 27.3.4-1. ILM-Transactions Field Descriptions (2 of 2)

Field Name	Data Type	Size	Entry	Description
From ECS Name	Char	30	System-supplied	The original Name of the machine with which the item is associated.
To ECS Name	Char	30	System-supplied	Name of the machine with which the item is associated.
From Location	Char	6	System-supplied	The original location of the EIN.
To Location	Char	6	System-supplied	The new location of the EIN.
From Building	Char	6	System-supplied	The original building where the EIN was located.
To Building	Char	6	System-supplied	The new building where the EIN is located.
From Room	Char	15	System-supplied	The original room where the EIN was located.
To Room	Char	15	System-supplied	The new room where the EIN is located.

Table 27.3.4-2. Procedures to Generate EIN Shipment Report

Perform	Action	Expected Result
Navigate to the ILM-Transaction Log Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-Transaction Log -> Search**	♦ ILM-Transaction Log form is displayed.
Run ECS Shipping Report	♦ Enter **SHP** into the Trans Type field. ♦ Use the Advance Search Bar to specify the Date-time period to report and click the search button. ♦ When the records are displayed, make sure to un-select the highlighted record by pressing the **Ctrl** and single click the highlighted item simutaneously. ♦ **Tools - > Reporting - > ECS Shipping Report - > Report - > Preview** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The ECS Shipping Report is displayed. See Figure 27.3.4-3 for the report layout.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Acrobat Reader – [~ECD9.pdf]

File   Edit   Document   View   Window   Help

Date: 7/24/03 4:41:36PM

ECS SHIPPING REPORT
Report Period: From 6/3/03 To 7/24/03
CONTRACT # NAS5 - 60000

Page 1 of 1

EIN	ECS NAME	MFR	DESCRIPTION	SERIAL NO	FROM	TO	Operator ID
00001881	XRPSERV	SUN	ULTRA 2 MODEL 1170	647F0973	EDF	EDF	ttran
00000127	CLE3IBM	IBM	RISC 6000 WORKSTATION	MS70122667441	EDF	LARC	ilmuser
C0001856	CLE3IBM	IBM	3.5 INCH FLOPPY DRIVE - 1.44 MB	L1PA3031294	LARC	PVC	ilmuser
C0001865	CLE3IBM	IBM	FDDI - ADAPTER	4140816	LARC	PVC	ilmuser
C0041911	NEWMAN	SUN	SINGLE CHANNEL ULTRA SCSI	9191755100	ASF	EDF	ttran
C0041913	NEWMAN	SUN	OPT INTERNAL 20GB 7200RPM EIDE HARD	3ECOWW8A	ASF	EDF	ttran
00006002	NEWMAN	STK	LIBRARY CONTROL UNIT	00810592	ASF	EDF	ttran

120%   1 of 1   11 × 8.5 in

Figure 27.3.4-3. ECS Shipping Report GUI

27-64

611-EMD-001

27.3.5 ILM-OEM Parts

Property Administrator uses the ILM-OEM Parts form (Figure 27.3.5-1) to maintain standardized information about manufacturer's parts. Part information must be recorded before they can be added to an inventory item.

Figure 27.3.5-1. ILM-OEM Parts GUI

Table 27.3.5-1 describes the fields on the ILM-OEM Parts form.

Table 27.3.5-1. ILM-OEM Parts Field Descriptions

Field Name	Data Type	Size	Entry	Description
Part No	Char	34	Required	Manufacturer's or vendor's part number for an item.
MFR	Char	6	Required	Code for the manufacturer of the item. The MFR code must exist in the ILM-Vendor-MFR form.
Hw-Sw Code	Char	2	Required	Code for classifying items according to source of maintenance costs. The Hw-Sw Code must exist in the ILM-HwSw Code Form.
Description	Char	60	Optional	Manufacturer's or vendor's description of the item.
Mod-Ver	Char	24	Optional	Model or version of the item.
Submitter	Char	30	System-supplied	The user whom created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user last modified the record.

27.3.6 ILM-Vendor-MFR GUI

The ILM-Vendor-MFR form (Figure 27.3.6-1) allows operators to define valid vendor codes for used with EIN records and Oem parts records. The operator enters or modifies the fields for this screen as required (see Table 27.3.6-1).

Figure 27.3.6-1. ILM-Vendor-MFR GUI

Table 27.3.6-1 describes the fields on the ILM-Vendor-MFR form.

Table 27.3.6-1. ILM-Vendor-MFR Field Descriptions

Field Name	Data Type	Size	Entry	Description
Vendor ID	Char	6	Required	Code for a vendor from whom items are purchased.
Vendor Name	Char	30	Required	Full name of a vendor from who items are purchased.
Submitter	Char	30	System-supplied	The user whom created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user last modified the record.

27.3.7 ILM-HwSw Codes GUI

Operators use this form (Figure 27.3.7-1) to maintain a standard set of codes for distinguishing items according to source of maintenance costs. These codes are associated with EIN items and are essential for grouping inventory items for reporting and browsing.

Figure 27.3.7-1. ILM-HwSw Codes GUI

Table 27.3.7-1 describes the fields on the ILM-HwSw Codes form.

Table 27.3.7-1. ILM-HwSw Codes Field Descriptions

Field Name	Data Type	Size	Entry	Description
Hw/Sw Code	Char	2	Required	Code for classifying items according to source of maintenance costs.
Description	Char	30	Required	Description for the Hardware/Software code.

611-EMD-001

27.3.8 ILM-Status Codes GUI

The ILM-Status Codes form (Figure 27.3.8-1) maintains a set of standardized codes for identifying valid inventory item states in the inventory and logistics life cycle.

Figure 27.3.8-1. ILM-Status Codes GUI

Table 27.3.8-1 describes the fields on the ILM-Status Codes form.

Table 27.3.8-1. ILM-Status Codes Field Descriptions

Field Name	Data Type	Size	Entry	Description
Item Status	Char	6	Required	Code for an inventory status for an item.
Description	Char	30	Required	Description for the code.

27.3.9 ILM-Maint Contract GUI

The ILM-Maint Contract form (Figures 27.3.9-1 and 27.3.9-2) provides the ability to track information about maintenance contracts with vendors and suppliers. The contract ID is the key field and should be the actual number that purchasing or the vendor assigns. The data entered here supports data entry for the ILM-EIN form (section 27.3.1). User may run a Maintenance Contract report that lists all EIN records that the individual maintenance contract covers. Table 27.3.9-2 provides instruction on how to run this report.

Figure 27.3.9-1. ILM-Maint Contract GUI (1 of 2)
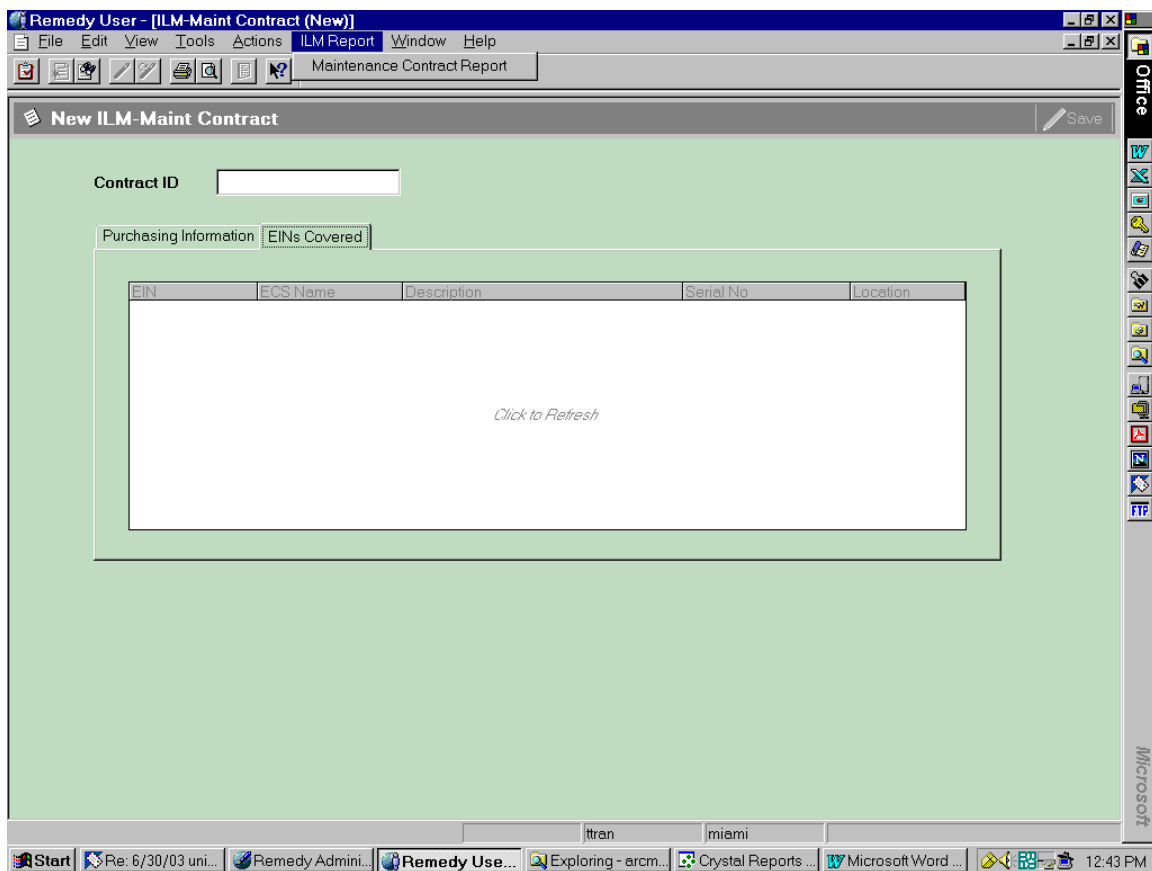
Figure 27.3.9-2. ILM-Maint Contract GUI (2 of 2)

Table 27.3.9-1 describes the field descriptions for the ILM-Maint Contract form.

Table 27.3.9-1. ILM-Maint Contract Field Descriptions

Field Name	Data Type	Size	Entry	Description
Contract ID	Char	10	Required	Identifier for the maintenance contract as assigned by purchasing or provided by the vendor.
PO Number	Char	10	Required	Purchase order number of the purchase order that procured the maintenance coverage.
Vendor ID	Char	6	Optional	Code for the vendor with whom the contract is placed.
Maint Vendor ID	Char	6	Optional	Code for the vendor whom will provide the services.
Start Date	Date		Optional	Date the contract is to become effective.
Expiration Date	Date		Required	Date the contract will expire.
Type of Support	Char	40	Optional	Type of support procured.
Comment	Char	60	Optional	Miscellaneous information specific to the maintenance contract.
EINs Covered	Table field		System-supplied	Field for displaying the EINs covered under the maintenance contract.
Submitter	Char	30	System-supplied	The user whom created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user last modified the record.

Table 27.3.9-2. Procedures to Generate Maintenance Contract Report

Perform	Action	Expected Result
Navigate to the ILM-Maint Contract Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-Maint Contract -> Search**	♦ ILM-Maint Contract form is displayed.
Run Maintenance Contract Report	♦ Find the Maintenance Contract of interest ♦ **ILM Report - > Maintenance Contract Report** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Maintenance Contract Report is displayed. See Figure 27.3.9-3 for the report layout.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Figure 27.3.9-3. Maintenance Contract Report GUI

27.3.10 ILM-Sites GUI

This form (Figure 27.3.10-1) allows operators to maintain a set of valid standard codes and descriptions for identifying ECS sites. Each code represents one site.

Figure 27.3.10-1. ILM-Sites GUI

Table 27.3.10-1 describes the fields on the ILM-Sites form.

Table 27.3.10-1. ILM-Sites Field Descriptions

Field Name	Data Type	Size	Entry	Description
Site	Char	6	Required	Code for an ECS site.
Description	Char	40	Required	Description of the site.

611-EMD-001

27.3.11   ILM-Inventory Location GUI

The form shown in Figure 27.3.11-1 is used to maintain information about ECS inventory locations. This standardized information is available to other screens and reports, which can access it by reference to a location.

Note: An important distinction is made in Remedy between an ECS site and an inventory location. Sites are officially designated by NASA and generally include the SMC, DAACs, and other official support installations. ECS Property Administrators designate inventory locations for purposes of property management. They are typically facilities or locales where inventory items are stored or installed, and more than one can exist at a site. Inventory locations are sometimes assigned the same names and codes as a site, but Remedy ILM treats the two as different entities.

Figure 27.3.11-1. ILM-Inventory Location GUI

Table 27.3.11-1 describes the fields on the ILM-Inventory Location form

Table 27.3.11-1. ILM-Inventory Location Field Descriptions

Field Name	Data Type	Size	Entry	Description
Location	Char	6	Required	Identifier for the inventory location where material can be found.
Building	Char	6	Optional	Building where the inventory items can be found.
Description	Char	30	Required	Description for the location id.
Site	Char	6	Required	Code for the ECS site hosting the inventory location.

27.4 Maintenance Management

Maintenance Work Orders (MWOs) are used for collecting downtime information for equipment subject to Reliability, Maintainability, and Availability (RMA) reporting as well as to identify equipment that has failed and/or been replaced during system maintenance. By way of a special feature available to the ILM-MWO and the ILM-MWO Line Item forms, operators can have the system update property records automatically based on the maintenance activities a work order describes. The following sections will describe maintenance work orders in detail.

27.4.1   ILM-MWO GUI

The ILM-MWO form (Figures 27.4.1-1 through 27.4.1-5) provides the ability to create and update maintenance work orders as maintenance activity proceeds and as additional information about the repair becomes known. It also has a special feature that updates property records on demand based on events and data described about a work order's line items.

Figure 27.4.1-1. ILM-MWO GUI – Parent Information

Figure 27.4.1-2. ILM-MWO GUI - Failure and Vendor Contact

Figure 27.4.1-3. ILM-MWO GUI – ALDT

Figure 27.4.1-4. ILM-MWO GUI - Total Down Time

Figure 27.4.1-5. ILM-MWO GUI - Failed & Replacement Components

Table 27.4.1-1 describes the fields on the ILM-MWO Form.

Table 27.4.1-1. ILM-MWO Field Descriptions (1 of 2)

Field Name	Data Type	Size	Entry	Description
Work Order No	Char	10	optional	Identifier for the work order. The work order no is prefixed with site first three letter.
MWO Status	Char	1	optional; O, A, F, or R	Code for the status of the work order. O = Open; A = Audit; F=Finish; R = Retired.
Priority	Char	1	optional	Code for the priority assigned to the work.
Parent EIN	Char	20	required	EIN for the parent item in an EIN structure.
System Serial Number	Char	30	system-supplied	Serial number of the item entered as Parent EIN.
ECS Name	Char	30	system-supplied from EIN record	Name of the machine with which the item is associated.
Part No	Char	34	system-supplied from EIN record	Manufacturer's part number for the item entered as Parent EIN.
Description	Char	60	system-supplied from EIN record	Manufacturer's description for the item entered as Parent EIN.
Location	Char	8	system-supplied from EIN record	Designator for the location where the item entered as Parent EIN is situated.
Room	Char	6	system-supplied from EIN record	Room in which the item entered as Parent EIN is situated.
Notification Date-Time	Date-Time		optional	The date and time problem was reported. This field is initialized with the current date and time but can be modified.
Failure Date-Time	Date-Time		optional	Date and time that the failure occurred. The earliest between the following: 1) the time the LMC (Local Maintenance Coordinator) is notified or 2) the time the hardware problem is first recognized.
Brief Description	Char	140	optional	A brief description of the problem and resolution.
Long Description	Char	1024	optional	A long description of the problem and resolution relevant to the maintenance event.

611-EMD-001

Table 27.4.1-1. ILM-MWO Field Descriptions (2 of 2)

Field Name	Data Type	Size	Entry	Description
Vendor Call Date-Time	Date-Time		optional	The date and time the maintenance vendor was called.
Vendor Initial Resp Date-Time	Date-Time		optional	Indicate the vendor initial response date and time to the service call.
Vendor Arrive Date-Time	Date-Time		optional	The date and time the maintenance vendor actually arrived to perform the repairs.
Vendor Complete Date-Time	Date-Time		optional	Date and time the repair was completed.
Vendor Contact Name	Char	30	optional	Vendor point of contact.
Vendor Reference	Char	20	optional	Identifier to the reference when contacting the vendor about the problem with the item.
ALDT Start Date-Time	Date-Time		optional	The date and time a delay in repairing the system began.
ALDT End Date-Time	Date-Time		optional	The date and time a delay in repairing the system ended.
ALDT Reason	Char	60	optional	A code for the reason a delay was encountered.
Start Restore Date-Time	Date-Time		optional	The date and time when start restoring the failed system.
End Restore Date-Time	Date-Time		optional	The date and time end restoring the system.
Total ALDT	Real	5.2	optional	Total Administrative logistic delay time (ALDT) Specified in hours.
Time To Repair	Real	5.2	optional	Time required to effect the repair. Specified in hours.
Switchover Time	Real	5.2	optional	Time required for system switch-over. Specified in hours.
Total Chargeable Down Time	Real	5.2	optional	Time to be charged for downtime. Specified in hours.
Submitter	Char	30	system-supplied	The user whom created the record.
Create Date	Date		system-supplied	Date the record was created.
Last Modified By	Char	30	system-supplied	The name of the user who last changed this record.
Modified Date	Date		system-supplied	The last date the record was modified.
Failed & Replacement Components	Table field		system-supplied	Field for displaying the failed and replacement components.

611-EMD-001

The following buttons/functions are unique to the ILM-MWO form:

- Add Fail/Replacement Component - provides access to the ILM-MWO Line Item Form (Figure 27.4.2-1) for adding data about components involved in individual maintenance actions.

- Process MWO Line Item - The Process MWO Line Item button provides a convenient, reliable, and efficient means for updating ILM property records based on information contained in ILM-MWO line item form. New EIN records are created as necessary, as are corresponding Part No and EIN structure records. Processing adds new items to the ECS inventory, archives those that have failed or been returned to the vendor, and re-assigns any that have been relocated or returned to stock. Additionally, items returned to a vendor are rendered obsolete with respect to their parent EINs and, of those that had failed, costs are transferred to their replacements. If Remedy is to update property records based on ILM-MWO line item data, line item records must specify values for Event Type and Maint Code. They determine the type of property record changes to be made. (See Table 27.4.2-2) Additionally, operators must supply a value for New Parent EIN if an item is designated for relocation. Other line item fields, such as Component EIN, Change Date, Replacement's EIN, New Location, and New Room, have special significance as well in that they influence which database records actually change.

Table 27.4.1-2. Procedure to Add a New Work Order

Perform	Action	Expected Result
Navigate to the ILM-MWO Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-MWO -> New**	ILM-MWO form is displayed and ready for the user to add a new MWO record.

Creating a new work order	Fill in the necessary information:
	♦ Work Order No – User may enter a Work Order No or leave this field blank. Leaving this field blank, the system will generate a Site's prefix work order no.
	♦ Select the Work Order Status from the MWO Status field. The default is "O" for Open.
	♦ Select a priority from the list of priority:
	    Enter a **1** for any malfunction that results in down time of a production system and immediate correction is needed.
	    Enter a **2** for any malfunction that impairs system performance but does not result in down time; however, may result in down time if system must be brought down to fix the problem.
	    Enter a **3** for any malfunction that will not result in system down time (e.g. minor flickering screen, key sticking, sticking mouse, etc…)
	♦ Enter **Parent EIN** number
	♦ Complete the Failure and Vendor Contact Information, ALDT, and Total Down Time sections.
	♦ Click on the **Save** icon on the tool bar to save the record.
		♦ When the Parent EIN is entered and the Parent EIN existed in the ILM-EIN form, the system will populate the Parent Information section with the information about the entered Parent EIN.
		♦ The system will generate the next Site's prefixed Work Order number after the record is saved. The newly created Work Order number is displayed in the Status bar on the bottom left of the form.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

611-EMD-001

Table 27.4.1-3. Procedures to Generate ILM-MWO Predefined Reports

Perform	Action	Expected Result
Navigate to the ILM-MWO Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-MWO -> Search** ♦ **Perform a Search for the records to be reported.**	♦ ILM-MWO form is displayed.
Run Work Order Verification Report	♦ Find the Work Order of Interest ♦ When the records are displayed, **Reports -> Work Order Verification** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon, or user may export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The Work Order Verification Report is displayed. Refer to Figure 27.4.1-6 for the report layout.
Run an RMA report and export into MS Excel	♦ Select the records of interest to perform the report on. ♦ When the record is displayed, make sure to un-select the highlighted record by pressing the **Ctrl** and single click the highlighted item simultaneously. ♦ **Tools -> Reporting -> Rma_Rpt -> Report -> Export To -> File** ♦ When the Report to File GUI (see Figure 27.3.1.1-3) is displayed. Select the appropriate directory to save the report in Enter a report file name **Save as Type** = All File (*.*) **Warning:** The report will not work if the **Save as Type** is anything else other than All File. Press the **Save** button ♦ Bring up MS Excel ♦ **File -> Open -> <Report Name>** ♦ When the Text Import Wizard GUI is displayed: **Delimiters –> Next** Select **Tab - > Next** Change the EIN, Parent EIN, Part No, and Serial No fields to Text. This will keep all the leading zeros. Click on the **Finish** button	♦ The RMA report is displayed. User may make any necessary ajustment to this report. Please refer to MS Excel help for any question on how to use MS Excel. Refer to Figure 27.4.1-7 for the report layout.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Acrobat Reader - [~IL28E.pdf]

File   Edit   Document   View   Window   Help

Date: 6/30/03   Work Order Verification Report   Page 1 of 1

Work Order No: EDF0000005   MWO Status: O   Priority:
Parent EIN: 00001881   ALDT 1
System Serial No: 647F0973
ECS Name: XRPSERV   Start Date-Time: End Date-Time:
Description: ULTRA 2 MODEL 1170   Reason:
MFR: SUN
Location: EDF   Room: 1073   ALDT 2
                                                                       Start Date-Time: End Date-Time:
Notification Date-Time: 6/26/03 0:00   Reason:
Failure Date-Time:
Vendor Call Date-Time:
Vendor Initial Resp Date-Time: Total ALDT:
Vendor Arrive Date-Time: Time to Repair:
Vendor Complete Date-Time: Switchover Time:
Maint Vendor: Total Chargable Down Time:
Vendor Contact Name:
Vendor Reference:

Brief Description:

Long Description:

Component EIN   Serial No   MFR   Description   Event Type   Maint Code   Change Date

120% |◄ ◄   1 of 1   ► ►| 11 x 8.5 in

Figure 27.4.1-6. Work Order Verification Report GUI

Work Order	MFR	DESCRIPTION OF SYSTEM DOWN	SYSTEM NAME	SITE	DATE/TIME Partially/Capabe	TOTAL Partially Capable TIME (HRS)	DATE/TIME SYSTEM DOWN	DATE RETURNED TO OPERATION	TOTAL DOWN TIME (HRS)	Restore Time	Problem Description & Solution	Notes
EDC0000509	SGI	RACK SERVER BASIC CHASIS	e0drg12	EDC	10/16/02 9:00		10/16/02 9:00	10/16/02 10:00	1.0	1.0	System board crashed due to bad node board.	Downtime was the result of troubleshooting and repair actions. System took an hour to restore but was operational.
LaR0000269	STK	9940A	STK Powerderho rn silo2	LaRC	10/15/02 8:00	28.0		10/16/02 12:00			Bad drive. Replace drive.	
EDC0000498	STK	Small Communications Rack	e0hippi1	EDC	09/30/02 5:15	224.7		09/30/02 16:00			Bad HIPPI fiber Channel card going to e0drg12.	No Down time associated because Ops had an alternative path GB router.
EDC0000500	SUN	Enterprise 4000 Enclosure 8-Slot Card Ca	e0sps04	EDC	09/09/02 2:45	8.3		09/09/02 11:00			32MB simm had parity errors causing box to reboot.	There are redundanct cpu boards with memory.

Figure 27.4.1-7. RMA Report GUI

27.4.2   ILM-MWO Line Item GUI

The ILM-MWO Line Item form (Figures 27.4.2-1 through 27.4.2-3) provides the ability for the Local Maintenance Coordinator (LMC) to identify equipment that has failed and/or been replaced during system maintenance. In general, a line item should be created for each EIN component that has failed, been replaced, or been added new. Note: line items can be created even if an EIN record does not exist for the component. Line items may only be created via the ILM-MWO form by pressing the "Add Fail/Replacement Component" button on the bottom of the form. Table 27.4.2-1 describes the fields on the ILM-MWO Line Item form.

Figure 27.4.2-1. ILM-MWO Line Item GUI (1 of 3)

Figure 27.4.2-2. ILM-MWO Line Item GUI (2 of 3)

Figure 27.4.2-3. ILM-MWO Line Item GUI (3 of 3)

Table 27.4.2-1 describes the fields on the ILM-MWO Line Item form.

Table 27.4-2-1. ILM-MWO Line Item Field Descriptions (1 of 2)

Field Name	Data Type	Size	Entry	Description
Work Order No	Char	10	System-supplied	Identifier for the work order.
Parent EIN	Char	20	System-supplied	EIN for the parent item in an EIN structure.
MWO Status	Char	1	System-supplied	Code for the status of the work order. O = Open; A = Audit; F=Finish; R = Retired.
ECS Name	Char	30	System-supplied	Name of the machine with which the item is associated.
Location	Char	8	System-supplied	Designator for the location where the item entered as Parent EIN is situated.
Component EIN	Char	20	Optional	Identifier for an item that is a child (component) of a parent EIN and the target of the maintenance event. If the field is left blank, the system will create an inventory number with a C-prefix for it automatically when the line item is processed.
Database Values				If the entered Component EIN exists in the ILM-EIN form, the system will populate this section with the data derived from the ILM-EIN form that describes the component EIN undergoing the maintenance activity.
Observed Values				User may enter information that describes the Component EIN in this section. If the Component EIN does not exist in the database, the component EIN will be added to the database using the information provided in the Observed Values section.
Part No	Char	34	Optional	Manufacturer's or vendor's part number for the item.
Description	Char	60	Optional	Manufacturer's or vendor's description of the item.
MFR	Char	6	Optional	Code used for the manufacturer of the item.
Mod-Ver	Char	24	Optional	Model or Version of the item.
Serial No	Char	30	Optional	Serial number of the item.
Item Status	Char	6	Optional	Code that designates the status of the item.
Hw-Sw Code	Char	2	Optional	Code for classifying items according to source of inventory.
GFE Num	Char	8	Optional	Gov't Furnished Equipment (GFE) number for the item.

Table 27.4.2-1. ILM-MWO Line Item Field Descriptions (2 of 2)

Field Name	Data Type	Size	Entry	Description
Event Type	Char	1	Required	Code identifying a type of maintenance event (N=new item installed; F=failed item replaced; R=serviceable item replaced).
Maint Code	Char	1	Required	Code designating the item's disposition. Property records are updated differently depending on the value entered. (R = Relocate, S = Stock, V = Vendor).
Change Date	Date		Required	Effective date of the configuration change.
Processed?	Char	1	System-supplied	Flag signifying whether or not the line item has been processed.
Replacement's EIN	Char	20	Optional	Identifier of the new item being used as a replacement. This field is used only for items that have failed and replaced by a new EIN.
New Parent EIN	Char	20	Optional, Required when the Event Type = R and Maint Code = R	EIN of the item to which the Component EIN is to be re-assigned. This field is applicable only to components that have failed or are being replaced (Event Type="F" or "R"), and are being relocated (Maint Code="R"). *The value must be supplied or the item will not get processed.*
New Location	Char	6	Optional	Code for the new inventory location to which the item is to be assigned.
New Building	Char	6	Optional	Building where the item is to be installed.
New Room	Char	15	Optional	Room where the item is to be installed.
Submitter	Char	30	System-supplied	The user whom created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user last modified the record.

Table 27.4.2-2 lists the appropriate combinations of Event Type and Maint Code and the effects on property records when processed.

611-EMD-001

Table 27.4.2-2. Effects on Property Records by MWO Line Item Processing (1 of 4)

Event Type	Maint Code	Situation/Property Record Updates

F (Failed)	S (Stock)	♦ When an item failed and returned to stock.
		EIN records:
		• Creates a record if one doesn't exist for the specified component EIN
		• For the specified component EIN:
		   • Clears its Parent EIN
		   • Clears its installation date
		   • Sets Item status to "F"
		   • Sets audit date to the Change Date
		   • Sets ECS name to "IN STOCK"
		   • Sets location, building, and room to new values.
		OEM part records:
		• Creates an OEM Part record if "observed values" for Part No, MFR, and Description are specified and the part record doesn't already exist
		EIN structure records:
		• Obsoletes the specified component EIN in EIN Structures where it is active. The structure is rendered inactive as of the specified Change Date
		Inventory transaction records:
		• Creates an entry for event of type "MFS" for the specified component

F (Failed)	V (Vendor)	♦ When an item failed and returned to the vendor.
		EIN records:
		• If the Component EIN field is blank, the system will not process the record and sets the Process field to X.
		• Creates a record if one doesn't exist for the specified component EIN
		• For the specified component EIN:
		   • Clears its Parent EIN
		   • Clears its installation date
		   • Sets item status to "X"
		   • Sets audit date to the Change Date
		   • Sets ECS name to "ARCHIVE"
		   • Sets location to "EDFARC" and clears building and room values.
		OEM part records:
		• Creates an OEM Part record if "observed values" for Part No, MFR, and Description are specified and the part record doesn't already exist
		EIN structure records:
		• Obsoletes the specified component EIN in EIN Structures where it is active. The structure is rendered inactive as of the specified Change Date
		Inventory transaction records:
		• Creates an entry for event of type "MFV" for the specified component

611-EMD-001

Table 27.4.2-2. Effects on Property Records by MWO Line Item Processing (2 of 4)

Event Type	Maint Code	Property Record Updates

N
(New)
	S
	(Stock)
		♦ When the replacement item is new and taken from stock.
		EIN records:
		• Creates a record if one doesn't exist for the specified component EIN
		• For the specified component EIN:
		    • Sets the Parent EIN to the MWO's Parent EIN
		    • Sets installation date to the Change Date
		    • Sets item status to "I"
		    • Sets audit date to the Change Date
		    • Sets ECS name to that of the Parent EIN specified for the MWO itself
		    • Sets location and room values to that of the Parent EIN specified for the MWO
		OEM part records:
		• Creates an OEM Part record if "observed values" for Part No, MFR, and Description are specified and the part record doesn't already exist
		EIN structure records:
		• Obsoletes the specified component EIN in EIN Structures where it is active, if any. The structure is rendered inactive as of the specified Change Date
		• Adds the EIN as a component of the item specified as a component of the MWO's Parent EIN. The structure is rendered active as of the Change Date specified
		
		Inventory transaction records:
		• Creates an entry for event of type "MNS" for the specified component

611-EMD-001

**Table 27.4.2-2. Effects on Property Records by
MWO Line Item Processing (3 of 4)**

Event Type	Maint Code	Property Record Updates

N
(New)
	V
	(Vendor)
		♦ When the replacement item is newed and came from the vendor.
		EIN records:
		• Creates a record if one doesn't exist for the specified component EIN
		• For the specified component EIN:
		  • Sets the Parent EIN to the MWO's Parent EIN
		  • Sets installation date to the Change Date
		  • Sets receive date to the Change Date
		  • Sets item status to "I"
		  • Sets audit date to the Change Date
		  • Sets ECS name to that of the Parent EIN specified for the MWO itself
		  • Sets location, building, and room values to that of the Parent EIN specified for the MWO itself
		  • If the component is replacing an EIN specified in a separate line item as a failed item being returned to the vendor and , copies the item cost from the EIN record for the failed item to the EIN record for the new item
		• For a failed item being replaced by the specified component EIN:
		  • Sets cost to 0
		
		OEM part records:
		• Creates an OEM Part record if "observed values" for Part No, MFR, and Description are specified and the part record doesn't already exist
		
		EIN structure records:
		• Obsoletes the specified component EIN in EIN Structures where it is active, if any. The structure is rendered inactive as of the specified Change Date
		• Adds the EIN as a component of the item specified as a component of the MWO's Parent EIN. The structure is rendered active as of the Change Date specified
		
		Inventory transaction records:
		• Creates an entry for event of type "MNV" for the specified component

Event Type	Maint Code	Property Record Updates

R
(Replaced)
	R
	(Relocate)
		♦ When an item is being relocated to a new machine.
		EIN records:
		• Creates a record if one doesn't exist for the specified component EIN
		• For the specified component EIN:
		   • Sets the Parent EIN to the New Parent EIN
		   • Sets the Installation Date to Change Date
		   • Sets item status to "I"
		   • Sets audit date to the Change Date
		   • Sets ECS name to the name of the new parent EIN
		   • Sets location, building, and room to that of the new parent EIN
		OEM part records:
		• Creates an OEM Part record if "observed values" for Part No, MFR, and Description are specified and the part record doesn't already exist
		EIN structure records:
		• Obsoletes the specified component EIN in EIN Structures where it is active. The structure is rendered inactive as of the specified Change Date
		• Adds the EIN as a component of the item specified as New Parent EIN. The structure is rendered active as of the specified Change Date
		Inventory transaction records:
		• Creates an entry for event of type "MRR" for the specified component

R
(Replaced)
	S
	(Stock)
		♦ When an item is being returned to stock.
		EIN records:
		• Creates a record if one doesn't exist for the specified component EIN
		• For the specified component EIN:
		   • Sets the Parent EIN to blank
		   • Clears its installation date
		   • Sets item status to "R"
		   • Sets audit date to the Change Date
		   • Sets ECS name to "IN STOCK"
		   • Sets location, building, and room to new values, if specified
		OEM part records:
		• Creates an OEM Part record if "observed values" for Part No, MFR, and Description are specified and the part record doesn't already exist
		EIN structure records:
		• Obsoletes the specified component EIN in EIN Structure where it is active. The structure is rendered inactive as of the specified Change Date
		Inventory transaction records:
		• Creates an entry for event of type "MRS" for the specified component

611-EMD-001

Table 27.4.2-3. Procedure to Add Work Order Line Items (1 of 3)

Perform	Action	Expected Result
Navigate to the ILM-MWO Line Item Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-MWO -> Search** ♦ Find the Work order No to add the line items ♦ When the Work order is displayed, click on the "Add Fail-Replacement Component" button on the bottom of the ILM-MWO form **Warning:** User **Must** use the ILM-MWO form to create new work order line items. The reason being is that the system needs to know the work order where the line item was generate from.	ILM-MWO Line Item form is displayed and ready for the user to add new work order line items.
Entering a failed item and returned to vendor	At the ILM-MWO Line Item form, ♦ Component EIN – Enter a component EIN if known – the system populates the "Database Values" section if the Component EIN existed in the ILM-EIN form. ♦ If the Component EIN does not exist in the database, complete the "Observed Values" section. ♦ Event Type – Enter an "**F**" for failed. ♦ Maint Code – Enter an "**V**" to specify that the failed component was returned to the vendor. ♦ Enter a Replacement EIN into the Replacement EIN field if known. ♦ Press the "Add Another LI to MWO" button.	♦ The system added the Failed component to the ILM-MWO Line Item form and is ready for user to add another line item to the work order. ♦ Refer to table 27.4.2-2 Event Type (F) and Maint Code (V) for detail description of property updates after the Property Administrator processes this line item.
Entering a failed item and return to stock	At the ILM-MWO Line Item form, ♦ Component EIN – Enter a component EIN if known – the system populates the "Database Values" section if the Component EIN existed in the ILM-EIN form. ♦ If the Component EIN does not exist in the database, complete the "Observed Values" section. ♦ Event Type – Enter an "**F**" for failed. ♦ Maint Code – Enter an "**S**" to specify that the failed component was returned to stock. ♦ Enter the new stock location values in the New Location, New Building, and New Room fields. ♦ Press the "Add Another LI to MWO" button.	♦ The system added the Failed component to the ILM-MWO Line Item form and is ready for user to add another line item to the work order. ♦ Refer to table 27.4.2-2 Event Type (F) and Maint Code (S) for detail description of property updates after the Property Administrator processes this line item.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.4.2-3. Procedure to Add Work Order Line Items (2 of 3)

Perform	Action	Expected Result
adding a component EIN to the MWO's Parent EIN structure where the Component EIN came from stock	At the ILM-MWO Line Item form, ♦ Component EIN – Enter a component EIN if known – the system populates the "Database Values" section if the Component EIN existed in the ILM-EIN form. ♦ If the Component EIN does not exist in the database, complete the "Observed Values" section. ♦ Event Type – Enter an "**N**" for failed. ♦ Maint Code – Enter an "**S**" to specify that the failed component came from stock. ♦ Press the "Add Another LI to MWO" button.	♦ The system added the Failed component to the ILM-MWO Line Item form and is ready for user to add another line item to the work order. ♦ Refer to table 27.4.2-2 Event Type (N) and Maint Code (S) for detail description of property updates after the Property Administrator processes this line item.
adding a new component EIN to the MWO's Parent EIN Structure where the Component EIN came from the vendor.	At the ILM-MWO Line Item form, ♦ Component EIN – Enter a component EIN if known – the system populates the "Database Values" section if the Component EIN existed in the ILM-EIN form. ♦ If the Component EIN does not exist in the database, complete the "Observed Values" section. ♦ Event Type – Enter an "**N**" for failed. ♦ Maint Code – Enter an "**V**" to specify that the failed component was returned to the vendor. ♦ Press the "Add Another LI to MWO" button.	♦ The system added the Failed component to the ILM-MWO Line Item form and is ready for user to add another line item to the work order. ♦ Refer to table 27.4.2-2 Event Type (N) and Maint Code (V) for detail description of property updates after the Property Administrator processes this line item.
Relocating a component to a new Parent EIN structure	At the ILM-MWO Line Item form, ♦ Component EIN – Enter a component EIN if known – the system populates the "Database Values" section if the Component EIN existed in the ILM-EIN form. ♦ If the Component EIN does not exist in the database, complete the "Observed Values" section. ♦ Event Type – Enter an "**R**" for relocate. ♦ Maint Code – Enter an "**R**" to specify that the component EIN is being relocated to a new Parent EIN. ♦ Enter the new Parent EIN into the New Parent EIN field. ♦ Press the "Add Another LI to MWO" button.	♦ The system added the Failed component to the ILM-MWO Line Item form and is ready for user to add another line item to the work order. ♦ Refer to table 27.4.2-2 Event Type (R) and Maint Code (R) for detail description of property updates after the Property Administrator processes this line item.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

611-EMD-001

Table 27.4.2-3. Procedure to Add Work Order Line Items (3 of 3)

Perform	Action	Expected Result
Relocating a component to stock	At the ILM-MWO Line Item form, ♦ Component EIN – Enter a component EIN if known – the system populates the "Database Values" section if the Component EIN existed in the ILM-EIN form. ♦ If the Component EIN does not exist in the database, complete the "Observed Values" section. ♦ Event Type – Enter an "**R**" for relocate. ♦ Maint Code – Enter an "**S**" to specify that the component EIN is being returned to stock. ♦ Enter the new stock location values in the New Location, New Building, and New Room fields. ♦ Press the "Add Another LI to MWO" button.	♦ The system added the Failed component to the ILM-MWO Line Item form and is ready for user to add another line item to the work order. ♦ Refer to table 27.4.2-2 Event Type (R) and Maint Code (S) for detail description of property updates after the Property Administrator processes this line item.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

27.5 License Management

Many software products used in ECS are licensed; that is, subject to conditions limiting how many users can run the product and where. Licenses take numerous forms. Nodelock licenses let users run the product, but only on a designated machine; counted nodelock licenses limit the number of users that can run the product on that machine. Floating licenses allow users to run a product from any machine in a network. They may limit the number of users that can run the product concurrently, the number of servers that can be used concurrently, the number of sites that can use the product, or any combination of the above. Licenses can apply to a named product, one or more of its features, one or more of its versions, and/or one or more types of platforms. Some vendors enforce these provisions through use of license keys, but ECS is accountable for adhering to licensing provisions whether vendors use keys or not.

When purchasing a product or obtaining an upgrade, engineering determines what licensing provisions are required. Depending on the product, license entitlements may appear as separate line items on purchase orders, or frequently not. (For example, purchased licensing provisions may be provided with the product; that is, not purchased separately.) License certificates (rights to certify) typically accompany software when it arrives and, in the case of operating system software, accompanies the computers themselves. These certificates describe the licensing provisions that were purchased and may carry an associated cost. Sometimes, the certificates include a license key, but usually they represent the right to obtain keys.

Multiple licenses are sometimes obtained from the product vendor under the provisions of a single license certificate. Each license would account for part of the rights-to-use under the

certificate. Conversely, individual licenses can consume rights-to-use from more than one certificate. Each unique license key implies a unique license, but not every license has a key.

Licenses are allocated to the sites and host machines where their keys are installed, and keyless licenses are allocated to where their software products are installed. This is not so much for property accounting (i.e., cost accounting), but to verify adherence to purchased licensing provisions and to identify where licenses are used in case rights-to-use must transfer elsewhere.

License rights-to-use are counted differently depending on the type of licenses purchased. Rights for nodelock license are allocated and counted by node and are consumed at the rate of one license per node. Floating license rights are allocated and counted based on number of users on a network rather than by specific machines, where the network is represented by a machine on which the license is installed. Floating license rights are consumed at the rate of number of users per license. Occasionally, a purchased entitlement covers a total number of users across a limited number of machines. In this case, rights are consumed at the rate of one license per node as well as number of users per license.

The following forms provide the software license administrator (SLA) capabilities to manage software licenses.

- ILM-License Products – to maintain standardized information about manufacturer's part numbers.

- ILM-License Entitlement – to maintain records of purchased rights-to-use for licensed software.

- ILM-License – to maintain records of software licenses obtained from vendors and maintains license allocation.

- ILM-License Mapping – manages the mapping of a license to purchased entitlements.

- ILM-Additional Host – identifies redundant or backup server machines on which the license will be installed

27.5.1 ILM-License Products GUI

This form (Figure 27.5.1-1) provides the SLA the ability to maintain standardized information about manufacturers' part numbers. Entitlement Part number must be recorded before they can be added to an entitlement or license via the ILM-Entitlement form or the ILM-License form.

Figure 27.5.1-1. ILM-License Products GUI

Table 27.5.1-1 describes the ILM-License Products form field definition.

Table 27.5.1-1. ILM-License Products Field Descriptions

Field Name	Data Type	Size	Entry	Description
Entitlement Part No	Char	34	Required	Manufacturer's or vendor's part number for the entitlement.
MFR	Char	6	Required	Code for the manufacturer from whom the item was purchased.
Version	Char	34	Optional	Version number of the part.
Description	Char	50	Required	Manufacturer's or vendor's description for the entitlement.
ECS Alias	Char	30	Optional	Common name used in ECS for the licensed product and all its versions and variants.
License Type	Char	16	Optional	Classification that distinguishes among licenses according to rules of use. Examples include: floating (limited number of concurrent users), nodelocked (limited to use on a single machine), project (unlimited use anywhere by individuals working on a certain project), site (unlimited use at a single site), etc.
Submitter	Char	30	System-supplied	The user that created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user who last modified the record.

611-EMD-001

27.5.2 ILM-License Entitlement GUI

Operators use the ILM-License Entitlement form (Figure 27.5.2-1, 2, & 3) to maintain records of purchased rights-to-use for licensed software, including how many node and user rights-to-use have been consumed, remain, and are under maintenance. Rights consumed and remaining are computed automatically based on the licenses mapped against it. Table 27.5.2-1 provides ILM-License Entitlement field descriptions.

Figure 27.5.2-1. ILM-License Entitlement GUI (1 of 3)

Figure 27.5.2-2. ILM-License Entitlement GUI (2 of 3)

Figure 27.5.2-3. ILM-License Entitlement GUI (3 of 3)

611-EMD-001

Table 27.5.2-1. ILM-Entitlement Field Descriptions

Field Name	Data Type	Size	Entry	Description
Entitlement ID	Char	10	System supplies	Identifier for a purchased license entitlement.
Entitlement Part No	Char	34	Required	Manufacturer's or vendor's part number for the entitlement.
MFR	Char	6	System supplied	Code for the manufacturer from whom the item was purchased. in the field above.
Description	Char	50	System supplied	Manufacturer's or vendor's description for the entitlement.
ECS Alias	Char	30	System supplied	Common name used in ECS for the licensed product and all its versions and variants.
License Type	Char	16	System supplied	Classification that distinguishes among licenses according to rules of use.
Version	Char	34	System supplied	Version number of the part.
Rights to Use (RTU)	Integer	8	Optional	Quantity of node or user rights-to-use authorized by this purchased entitlement.
Node/User Allocated	Integer	8	System supplied	Quantity of node or user rights under the license entitlement currently allocated by licenses mapped to the entitlement. This value is calculated by the system and reflects the total number of allocations of those licenses.
Node/User Remaining	Integer	8	System supplied	Quantity of node or user rights under a license entitlement not yet consumed by the mapping of licenses to the entitlement.
Node/User Under Maint	Integer	8	System supplied	Quantity of node or user rights-to-use currently under maintenance.
Vendor ID	Char	6	Optional	Code for the vendor from whom the item was purchased.
Receive Date	Date		Optional	Date item was received from vendor.
Maint Contract ID	Char	10	Optional	Identifier for the Maintenance Contract under which the item is covered.
Maint Exp Date	Date		Optional	Date the maintenance contract expired.
Owner	Char	10	Optional	The owner of the entitlement.
Comment	Char	30	Optional	Miscellaneous information specific to the item.
Licenses	Page			This table lists any license(s) the license entitlement is mapped to and how many user rights-to-use was assigned to the license id.
Submitter	Char	30	System-supplied	The user whom created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user who last modified the record.

611-EMD-001

The following buttons are unique to this form:

- Add New Part – Activates the ILM-License Products form. This allows the operator to add new parts.
- Add New License – displays the ILM-License form (Figure 27.5.3) to allow the SLA to add new licenses.

Table 27.5.2-2 provides instruction on how to add a new License Entitlement.

Table 27.5.2-2. Procedure to Add New License Entitlement

Perform	Action	Expected Result
Navigate to the ILM-License Entitlement Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-License Entitlement -> New**	ILM-License Entitlement form is displayed and ready for the user to add new license entitlement.
Defining a new license entitlement	Filling in the necessary information, ♦ Entitlement ID – Leave this field blank. The system will generate the next prefixed "LE" number when user saves the record. ♦ Complete the Part information section. Note. The Entitlement Part No field's label is **Bolded**. Therefore, you must enter a value into the Entitlement Part No field. The Entitlement Part No must exist in the ILM-License Products form. If not, the system will display an error message and will not except the value. ♦ If the license entitlement is a Node lock licenese, complete the Node Rights-To-Use section. ♦ If the license entitlement is a floating license, complete the User Rights-To-Use section. ♦ Complete the Purchasing & Maintenance Info section. ♦ Press the **Save** icon to save the record.	♦ The system will generate the next prefixed "LE" licenes entitlement number and saves the new record in the database.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.5.2-3 provides instruction on how to run the License Entitlement status report that was specially designed for this form. This report lists the status of current license entitlements for licensed software products, sorted by software product, version, and license type.

611-EMD-001

Table 27.5.2-2. Procedures to Generate ILM-License Entitlement Predefined Reports

Perform	Action	Expected Result
Navigate to the ILM-License Entitlement Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-License Entitlement -> Search**	♦ ILM-License Entitlement form is displayed.
Run License Entitlement Status Report	♦ Find the records of interest ♦ When the records are displayed, **Tools - > Reporting -> License Ent Status Report** ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The License Entitlement Status Report is displayed. See Figure 27.5.2-4 for the report layout.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Acrobat Reader - [~Li2A2.pdf]

File   Edit   Document   View   Window   Help

Date: 6/30/03   License Entitlement Status Report   Page 1 of 1

AUTOEXPERT

LICENSE TYPE	ENT ID	DESCRIPTION	VENDOR	PO Number	MAINT CONTRACT	EXP DATE	USER RTU	URTU REM	USER MNT	NODE RTU	NRTU REM	NRTU MNT
NODELOCK	LE000024	AUTOEXPERT-HIGH AVAILABILITY SERVER OPT					0	0	0	20	19	0
NODELOCK	LE000025	AUTOEXPERT-HIGH AVAILABILITY SERVER OPT					0	0	0	20	19	0

USER RTU - Totoal User Rights-To-Use purchased.

URTU REM - Number of User Rights-To-Use remaining.

URTU MNT -Number of User Rights-To-Use having maintenance.

NODE RTU - Total Node Rights-To-Use purchased.

NODE REM - Number of node Rights-To_Use remaining

NODE MNT - Number of node Rights-To-Use having maintenance.

120%   1 of 1   11 x 8.5 in

Figure 27.5.2-4. License Entitlement Status Report GUI

27.5.3   ILM-License GUI

The ILM-License form (Figures 27.5.3-1 and 27.5.3-2) maintains records of software licenses obtained from vendors. This form also maintains records about the hosts and sites to which the licenses have been allocated. Licenses must be mapped to purchased license entitlements so that consumption of license rights can be tracked.

A license is a euphemism for the rights granted a number of user to operate a software product or one or more of the product's versions or features concurrently on certain machines. These rights are often encoded in a license "key", but not all products employ such keys.

Figure 27.5.3-1. ILM-License GUI (1 of 2)

Figure 27.5.3-2. ILM-License GUI (2 of 2)

Table 27.5.3-1 describes the field definition for the ILM-License form, and table 27.5.3-2 and 27.5.3-3 provide instructions on how to create a new license and how to map a license to entitlement.

Table 27.5.3-1. ILM-License Field Descriptions (1 of 2)

Field Name	Data Type	Size	Entry	Description
License ID	Char	10	System Supplies	Unique designator for a license.
Entitlement Part No	Char	34	Required	Manufacturer's or vendor's part number for the entitlement.
MFR	Char	6	System supplied	Code for the manufacturer from whom the item was purchased.
Description	Char	50	System supplied	Manufacturer's or vendor's description for the entitlement.
ECS Alias	Char	30	System supplied	Common name used in ECS for the licensed product and all its versions and variants.
Receive Date	Date		Optional	Date the license key and/or data arrived.
Version	Char	34	System supplied	Version number of the part.
Platforms	Char	15	Optional	One or more codes for the types of machines to which the license applies (e.g., Sun, SGI, PC, etc.)
Serial No	Char	30	Optional	Vendor-supplied serial number for the license or the product being licensed.
License Type	Char	16	System supplied	Classification that distinguishes among licenses according to rules of use.
Features	Char	54	Optional	Name(s) of one or more features of the licensed product that are covered by the license.
License Manager	Char	12	Optional	Technology employed in managing the license on-line (e.g., flexlm, proprietary, etc.)

611-EMD-001

Table 27.5.3-1. ILM-License Field Descriptions (2 of 2)

Field Name	Data Type	Size	Entry	Description
License Key	Char	50	Optional	Char of alphanumeric characters that represent the provisions for a license in an encoded form.
Host Name	Char	30	Optional	ECS Name of a machine to which the license is allocated.
Host ID	Char	20	Optional	Host id of the license server machine supplied to the vendor when requesting the license. Allocations of licenses to machines are accomplished via the License Allocation Manager screen.
Site	Char	6	Optional	Code for the site to which the license is allocated.
Lic Key User RTU	Integer	8	Optional	Number of users authorized by the license to run the licensed product concurrently on a single network. This value limits the user rights-to-use that can be recorded in the license's allocation records.
Password	Char	20	Optional	Password supplied along with the license key by the vendor.
Key Exp Date	Date		Optional	Date on which the license key is no longer usable.
Comment	Char	60	Optional	Comment to be stored in the record.
Submitter	Char	30	System-supplied	The user whom created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user last modified the record.

The following buttons are unique to this form:

- Add New Part – Activates the ILM-License Products form. This allows the operator to add new parts.
- License-Entitlement Mapping - activates the ILM-License Mapping form that allows the SLA to map the license to the purchased entitlement.
- Add Additional Host – activates the ILM-Additional host form to allow the SLA to add redundant host or backup server to the license.

Table 27.5.3-2. Procedure to Add New License and Allocate It to a Machine
(1 of 2)

Perform	Action	Expected Result
Navigate to the ILM-License Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-License -> New** Or ♦ While the ILM-License Entitlement form is displayed, press the "Add New License" button on the bottom of the form.	ILM-License form is displayed and ready for the user to add new license.
Defining a new license and allocate the license to a host	Filling in the necessary information, ♦ License ID – Leave this field blank. The system will generate the next prefixed "L" number when user saves the record. ♦ Complete the Part information section. Note. The Entitlement Part No field's label is **Bolded**. Therefore, you must enter a value into the Entitlement Part No field. The Entitlement Part No must exist in the ILM-License Products form. If not, the system will display an error message and will not except the value. ♦ License Key – Enter the license key that the vendor provided. ♦ Enter Host name, Host ID, and Site where the license is allocated. ♦ Complete the rest of the information in the License Key section. ♦ Press the **Save** icon to save the record.	♦ The system will generate the next prefixed "L" licenes entitlement number and saves the new record in the database. ♦ The system displays the ILM-License Mapping form (Figure 27.5.4-1) with the newly created license information that just created and ready for user to map the license to the puchased entitlement.
Mapping the new license to a puchased entitlement	When the **Save** button is pressed, the system displays the ILM-License Mapping form with the data about the newly created license record created in the step above. Enter the following information to complete mapping the license to a purchase entitlement: ♦ Entitlement ID – Enter an Entitlement ID where this license ID should be mapped against. ♦ If the license type is a Node lock license, enter "**1**" to allocate 1 node lock license to this license ID. ♦ If the license type is a floating license, enter the number of user rights-to-use allocated to this license ID. The number of User rights-to-use cannot exceed the number user rights-to-use assigned to the license ID. ♦ Press the **Save** icon to save the record. ♦ Close the ILM-License Mapping form.	♦ If the license type is a Node Lock license, the system will decrement 1 node rights-to-use remaining from the Entitlement ID and increments 1 from the Node rights-to-use allocated. ♦ If the license type is a floating license, the system will decrement 1 from the user rights-to-use remaining from the Entitlement ID and increments 1 from the user rights-to-use allocated.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

Table 27.5.3-2. Procedure to Add New License and Allocate It to a Machine (2 of 2)

Perform	Action	Expected Result
Adding additional host to the license	while the ILM-License form is on the displayed, ♦ Find the license of interest ♦ Press the "Add Adiitional Host" button on the bottom of the form to add additional host to the license ID. ♦ Enter the additional Host Name, Host ID, and Site. ♦ Press the **Save** icon to finish adding additional host to the license, OR ♦ Press the "Add Additional Host to License" button to add additional hosts to the license ID. ♦ Close the ILM-Additional Host form when finished adding all the additional hosts.	♦ The ILM-Additional Host form is displayed. See section 27.5.5 for more information about the ILM-Additional Host form. ♦ Additional hosts are added to the license.

Note: To move to the next field, use either the TAB key or the Mouse. The Enter key will not move the cursor to the next field.

More over, there are two pre-defined reports that were designed specially for this form. The license allocation by host report lists license allocations, sorted by host name and ECS part alias and the license allocation by product report lists license allocations for licensed software products, sorted by product, version, and host name. Table 27.5.3-4 provides instructions on how to run these reports. Refer to Figure 27.5.3-4 and 27.5.3-5 for the report layouts.

Table 27.5.3-3. Procedures to Generate ILM-License Predefined Reports

Perform	Action	Expected Result
Navigate to the ILM-License Form	At the Remedy-ILM PC, ♦ **File -> Open -> ILM-License -> Search**	♦ ILM-License Entitlement form is displayed.
Run License Allocation by Host Report	**Tools - > Reports -> License Allocation By Host** ♦ When the ILM-DIA-Lic Report form is displayed (see Figure 27.5.3-3), select or enter a site name, and/or enter the ECS Alias. Leaving the Site and the ECS Alias values blank will select all licenses in the database. ♦ Press the "Run Report" button to start running the report. ♦ Press on the "Report Preview" to bring up the report layout. ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The License Allocation by Host Report is displayed. See Figure 27.5.3-4 for the report layout.
Run License Allocation by product Report	**Tools - > Reports -> License Allocation By Product** ♦ When the ILM-DIA-Lic Report form is displayed (see Figure 27.5.3-3), enter the ECS Alias. Leaving the Site ECS Alias values blank will select all licenses in the database. ♦ Press the "Run Report" to start running the report. ♦ Press on the "Report Preview" to bring up the report layout. ♦ When the report is displayed, user has the option to either print the report by pressing the print icon or export the report into a file by pressing the Export Icon next to the print icon. ♦ Close the report preview.	♦ The License Allocation by Product Report is displayed. See Figure 27.5.3-5 for the report layout.

611-EMD-001
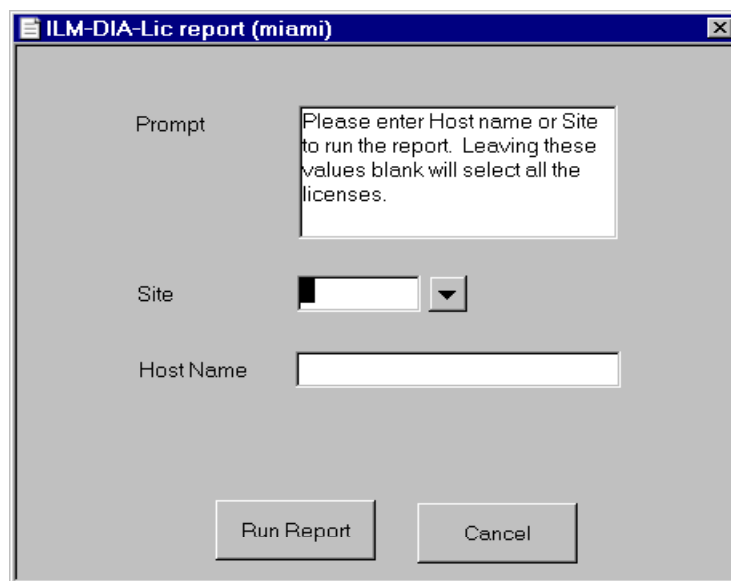
Figure 27.5.3-3. ILM-DIA-Lic Report GUI

Figure 27.5.3-4. License Allocation By Host Report GUI

Acrobat Reader - [~IL2BA.pdf]

File Edit Document View Window Help

Date: 6/30/03 1:40:00PM   License Allocations By Product   Page 1 of 2

CLEARCASE   Version: Features:

License_ID	Host_Name	Host_ID	License_key	Key_Exp_Date	Platforms	User_RTU
L0000450	t1mss04	807fe113	387d2a2e.01ee9020.02	10/31/02		5

License_ID	Host_Name	Host_ID	License_key	Key_Exp_Date	Platforms	User_RTU
L0000451	n0mss02	8080579b7	387d2a01.03aa6d75.02	10/31/02		10

License_ID	Host_Name	Host_ID	License_key	Key_Exp_Date	Platforms	User_RTU
L0000452	p0mss02	8080bfb7	387e702e.79f1f224.02	10/31/02		2

License_ID	Host_Name	Host_ID	License_key	Key_Exp_Date	Platforms	User_RTU
L0000453	p0mss02	8080bfb7	387e7228.528a5aad.02	10/31/02		15

License_ID	Host_Name	Host_ID	License_key	Key_Exp_Date	Platforms	User_RTU
L0000454	p0mss02	8080bfb7	387e717b.f4b05dab.02	10/31/02		6

License_ID	Host_Name	Host_ID	License_key	Key_Exp_Date	Platforms	User_RTU
L0000455	p0mss02	8080bfb7	387e70be.9600a857.02	10/31/02		5

License_ID	Host_Name	Host_ID	License_key	Key_Exp_Date	Platforms	User_RTU
L0000456	m0mss02	8080fd39	387d28fe.3da6012e.02	10/31/02		1

License_ID	Host_Name	Host_ID	License_key	Key_Exp_Date	Platforms	User_RTU
L0000457	g0mss02	8080c9e8	387d287f.95f02f0b.02	10/31/02		9

120%   1 of 2   11 x 8.5 in

Figure 27.5.3-5. License Allocation By Product Report GUI

27.5.4 ILM-License Mapping GUI

The ILM-License Mapping form (Figure 27.5.4-1) manages the mapping of a license to purchased entitlements and specifies how many node and/or user rights-to-use the license is consuming from each. The form ensures that:

a) The rights-to-use attributed to an entitlement do not exceed the entitlement's rights remaining;

b) The sum of the rights being attributed to all entitlements do not exceed the rights-to-use for the license.

Figure 27.5.4-1. ILM-License Mapping GUI

Table 27.5.4-1 describes the fields on the ILM-License Mapping form.

611-EMD-001

Table 27.5.4-1. ILM-License Mapping Field Descriptions

Field Name	Data Type	Size	Entry	Description
License ID	Char	10	System Supplies	Unique designator for a license.
Entitlement ID	Char	10	Required	Identifier for a purchased license entitlement.
ECS Alias	Char	40	System supplied	Common name used in ECS for the licensed product and all its versions and variants.
License Type	Char	16	System supplied	Classification that distinguishes among licenses according to rules of use.
MFR	Char	6	System supplied	Code for the manufacturer from whom the item was purchased. in the field above.
Lic Key User Rtu	Numeric	8	System supplied	Number of users authorized by the license to run the licensed product concurrently on a single network. This value limits the user rights-to-use that can be recorded in the license's allocation records.
Remain	Numeric	8	System supplied	Quantity of node or user rights under a license entitlement not yet consumed by the mapping of licenses to the entitlement.
Allocated	Numeric	8	Optional	Number of node or user rights-to-use to be counted under the entitlement as having been consumed by the license. The value may not exceed the current value plus the rights remaining under the entitlement.
Submitter	Char	30	System-supplied	The user that created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user who last modified the record.

611-EMD-001

27.5.5 ILM-Additional Host GUI

The ILM-Additional Host form (Figure 27.5.5-1) is used for maintaining records about backup or redundant license servers for machines to which a license has been allocated. Identifying additional hosts has no effect on calculations of entitlements node or user rights-to-use consumed or remaining, but is useful for tracking where redundant licenses are supposed to be or may be installed.

Figure 27.5.5-1. ILM-Additional Host GUI

Table 27.5.5-1 describes the fields on the ILM-Additional Host form.

Table 27.5.5-1. ILM-Additional Host Field Descriptions

Field Name	Data Type	Size	Entry	Description
License ID	Char	10	System Supplied	Unique designator for a license. Derived from the displayed license.
ECS Alias	Char	30	System Supplied	Common name used in ECS for the licensed product and all its versions and variants.
Host Name	Char	30	Optional	ECS name of a machine that is a backup or redundant license server for the one to which the license is principally allocated.
Host ID	Char	20	Optional	Host id of a machine that is a backup or redundant license server for the one to which the license is principally allocated.
Site	Char	6	Optional	Code for the site to which the license is allocated.
Submitter	Char	30	System-supplied	The user that created the record.
Create Date	Date		System-supplied	Date the record was created.
Last Modified By	Char	30	System-supplied	The user who last modified the record.

27.6 System Administrator Functions

27.6.1   ILM-System Parameters

The ILM-System Parameters form (Figure 27.6.1-1) is for maintaining system-wide Remedy-ILM parameters and is principally used for initializing certain identifier fields.

Several fields have particular significance for ILM. The Site ID field contains the code for the ECS site where the operator's copy of Remedy is installed. The field is interrogated by ILM processes that have to determine which assets belong to the local site.

Figure 27.6.1-1. ILM-System Parameters GUI

Table 27.6.1-1 describes the fields on the ILM-System Parameters form.

Table 27.6.1-1. ILM-System Parameters Field Descriptions

Field Name	Data Type	Size	Entry	Description
Site	Char	6	Required	Code that identifies the ECS site where this Remedy system is installed.
Last EIN	Char	20	System-supplied, but modifiable	Code used in determining the next sequentially-available identifier when assigning EIN numbers automatically.
Last MWO	Char	10	System-supplied, but modifiable	The last MWO number used.
Last Entitlement ID	Char	10	System-supplied, but modifiable	The last entitlement id used.
Last License ID	Char	10	System-supplied, but modifiable	The last license id used.

27.6.2   Inter-Site Data Exchange

Inter-site data exchange processes are executed on a daily basis to transfer EDF ILM data to other locations and to retrieve Maintenance Work Order data from the DAACs. Remedy export specific data to files, and supporting scripts/applications distribute these files to remote sites. These files contain centrally-managed ILM records that have changed since the last time this function was run. Only records about inventory items at the site are to be exported.

Remedy analyzes the following forms to determine what data changed and which records were affected: EIN, EIN Structure, OEM Parts, Vendor-MFR, Maint Contract, hardware_software codes, Status Code, MWO, MWO Line Item, License Products, License Entitlement, License, License Mapping, and Additional Host. Remedy selects changed records by site, then dumps their data into files compatible with Remedy's Import utility. Data files are combined into one tar file per site. Each tar file is given a name that identifies the date and time the export was done, the origination site, and the destination site. These files are placed in the export directory at the EDF.

27.6.3   DAAC Data Export

DAACs data export functions are similar to EDF. The biggest different is DAACs export only the maintenance work orders data changes. The two forms that contain maintenance work order data are the ILM-MWO and ILM-MWO Line Item forms.

Remedy analyses the ILM-MWO and the ILM-MWO Line Item forms to determine what data changed and which records were affected. It copies those records to files that are compatible with the Remedy's Import records utility. These two files are tarred into one tar file. The tarred file is given a name that identifies the date and time the export was done, the origination site, and forEDF. The exported file is placed in the export directory at the local site.

27.6.4   DAAC Data Import

Data import is done on a routine basis at each site. Remedy performs the data import at a specified time interval. At 1:00 AM every working day, the DAACs use Putty's pscp application to secure copy the exported file from the EDF export directory to the DAAC's Remedy PCs. When finished copying the file to the local Remedy PC, the DAACs move the EDF exported files to the EDF export archive directory. At the local Remedy PC, the copied file is untarred by using Winzip and then imported into the appropriate forms by using Remedy's Import utility.

27.6.4.1   EDF Data Import

EDF follows the same processes to import maintenance work order changes from the DAACs. At 3:00 AM every working day, EDF's Remedy PC uses Putty's pscp to secure copy the DAACs exported file from the DAAC's export directory to the EDF's Remedy PC. The import script uses Winzip to untar the file and finally imports the work order changes into Remedy-ILM at the EDF.

27.6.5   User GUI

The User form, shown in Figure 27.6.5-1, is used by the administrator to add, modify, or remove users from the Action Request (AR) System. The "User" form is used in conjunction with the "Group" form to provide users with permissions ultimately determining which operations individual users can perform and which forms and fields they can access. For more information on the "User" form and the AR System access control, refer to the Action Request System Server Administrator's Guide.

611-EMD-001

Figure 27.6.5-1. User GUI

Table 27.6.5-1. User Form Field Descriptions

Field Name	Data Type	Size	Entry	Description
Entry-Id	Character	15	System generated	Entry ID of user
Status	Selection	*	Required	Is user current or not as shown by the "current" button
License Type	Selection	*	Required	What type of license does this user have? (e.g., read, fixed, floating)
Login name	Character	30	Required	Login name of user
Password	Character	30	Optional	Password of User
Email Address	Character	255	Required	E-mail address of User
Group list	Character	255	Optional	Groups to which the user belongs
Full Name	Character	128	Required	Full Name of User
Phone Number	Character	55	Required	Phone Number of User
Home DAAC	Character	55	Required	Home DAAC of User
Default Notify Mechanism	Selection	*	Optional	Notification method (e.g., None, Notifier, and Email buttons)
Full Text License Type	Selection		Required	Not applicable
Creator	Character	30	Required	Person who created the user account
Create-date	Date/Time	17	System generated	Date and time the entry was created at the present site (mm/dd/yy hh:mm:ss)
Last-modified-by	Character	30	System generated	User ID of person that last modified the user entry
Modified-date	Date/Time	17	System generated	Date and time of last modification to user entry (mm/dd/yy hh:mm:ss)

*Note, the size of a field with a "selection" data type can vary and the size is automatically adjusted to the size of the item selected from the selection list.

27.6.6 Remedy's Admin Tool GUI

The Remedy Administrator Tool is the tool one uses to create, modify, and delete Remedy objects (e.g. forms and menus). Figure 27.6.6-1 shows the main Administrator Tool GUI and its starting screen, the server window, and the workflow objects categories.
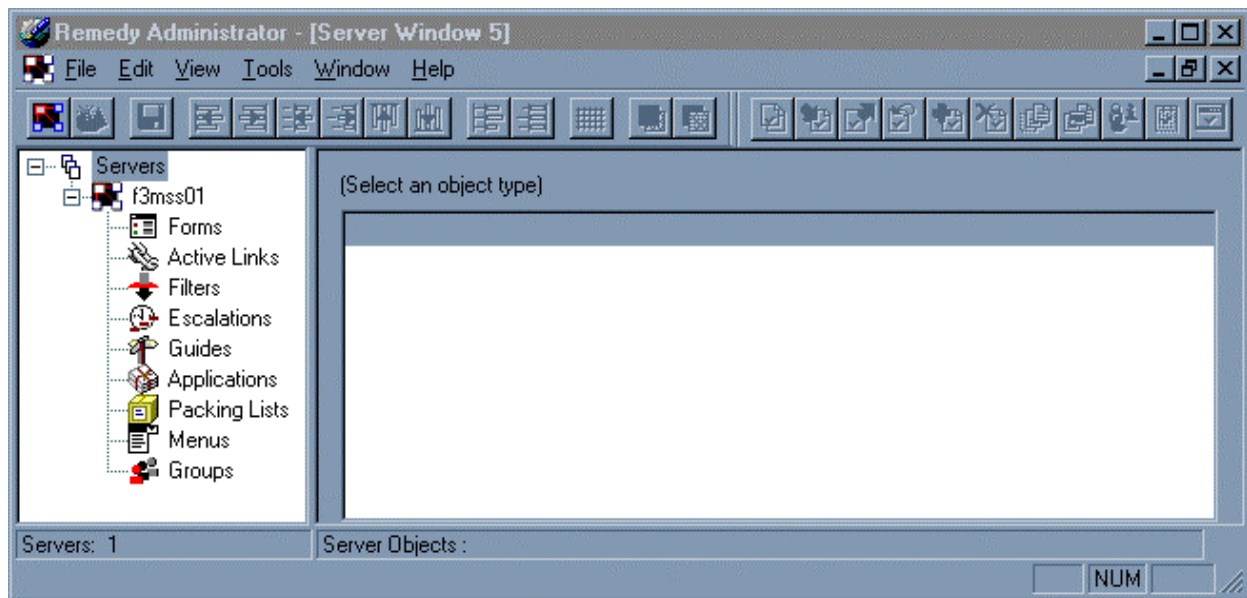
Figure 27.6.6-1. Admin Tool GUI

Table 27.6.6-1 provides a description of the Admin Tool GUI Workflow objects. For more information on these options, see *Remedy's Action Request System Administrator's Guide*, Vol. 1, Chapter 4, and/or the Remedy Administrator Tutorial using the Administrator Tool's Help menu.

Table 27.6.6-1. Admin Tool GUI, Workflow Object Descriptions

Workflow Object	Size	Entry	Description
Forms	Variable	System Generated	List of available forms.
Active links	Variable	System Generated	List of available active links.
Filters	Variable	System Generated	List of available filters.
Escalations	Variable	System Generated	List of available escalations.
Guides	Variable	System Generated	List of available guides.
Applications	Variable	System Generated	List of available applications.
Packing lists	Variable	System Generated	List of available packing lists.
Menus	Variable	System Generated	List of available menus.
Groups	Variable	System Generated	List of available groups.

27.6.7 Databases

Remedy's Action Request System uses a Sybase database called AR System. Tables and columns are created, modified, and deleted when forms are built and edited. This is all done automatically and is invisible to the user.

27.6.8 Special Constraints

Note that while ILM forms are open to all operators, and operators have view privileges to the user form, only system administrators have the ability to modify forms and tools presented in this section. Privileges are set according to DAAC policy.

27.6.9 Event and Error Messages

For Remedy's Action Request System's system messages see the *Action Request System Error Messages Guide*.

Table 27.6.9-1 lists non-system failure related messages which appear on the operator's screen.

Table 27.6.9-1. Non System-Failure Related Error Messages (1 of 5)

Error Message Char	Cause	Action
Inventory Management		
Parent EIN does not exist. Enter the correct Parent EIN.	Attempting to associate a component EIN to a Parent EIN that does not exist.	Enter the correct Parent EIN.
Parent EIN, $Parent EIN$, is a component. Please enter the correct Parent EIN number.	Attempting to associate a component EIN to a Parent EIN that is a component to another EIN Structure.	Enter the correct Parent EIN.
Parent EIN, $Parent EIN$, is not hardware. Enter the correct Parent EIN!	Attempting to associate a component EIN to a Parent EIN that is not hardware.	Enter the correct Parent EIN.
Part not found. Add new part into the part table or enter the correct part no.	Attempting to enter a part no that does not exist in the database.	Enter the correct Part No into the ILM-OEM Parts form or enter the correct part no.
Part Number, $Part No$, already exists. Enter the correct Part Number!	Attempting to add a new Part No enter the ILM-Oem Parts form where the Part No already existed in the form.	Enter the correct Part number.
EIN is a component to Parent EIN. Update the Parent EIN's ECS name instead.	Attempting to update an ECS Name of a component EIN.	Update the Parent EIN's ECS Name.
New Parent EIN is the same as the old Parent EIN. Enter the correct new Parent EIN.	Attempting to relocate to relocate an item to the same Parent EIN.	Enter the correct New Parent EIN.
EIN already existed. Enter the correct EIN.	Attempting to create a new EIN that is already existed in the database.	Enter the correct EIN number.

611-EMD-001

Table 27.6.9-1. Non System-Failure Related Error Messages (2 of 5)

Error Message Char	Cause	Action
A record for Location, $Location$, with Building, $Building$, already exists. Enter the correct Location and its associated Building!	Attempting to create a new location in the ILM-Inventory location form where the combination of location and building values already existed in the form.	Do not enter the new location and use the existed one.
EIN (EIN) is a Parent to EIN ($EIN is Parent$). Cannot assign a Parent to another Parent structure.	Attempting to assign a Parent EIN as a component to an EIN structure.	Verify the Component EIN and the Parent EIN.
Audit Date ($Audit Date$) cannot be greater than current date. Enter the correct audit date.	Attempting to update the audit date to a date greater than the current date.	Enter the correct audit date.
Receive Date ($Receive Date$) cannot be greater than the current date ($DATE$). Enter the correct receive date.	Attempting to update the receive date to a date greater than the current date.	Enter the correct receive date.
Vendor ID, $Vendor ID$, already exists. Enter the correct Vendor ID!	Attempting to add a new vendor id that already existed in the ILM-Vendor-MFR form.	Use the existing Vendor ID if the vendor name is the same. If the vendor name is not the same, use another code to define the new vendor.
Site, $Site$, already exists. Enter the correct Site!	Attempting to add a site code that already existed in the ILM-Site form.	Use the existing site code.
Item Status, $Item Status$, already exists. Enter the correct Item Status!	Attempting to add a new item status that already existed in the ILM-Status Code form.	Enter the correct item status.
EIN Transactions		
Parent EIN field is a required. Enter a Parent EIN value into the Parent EIN field.	Attempting to perform an EIN transacion where the Parent EIN value is not provided in the Parent EIN field.	Enter the correct Parent EIN value into the Parent EIN.
Parent EIN ($Parent EIN$) does not exist. Enter the correct Parent EIN.	Attempting to perform an EIN transaction on the Parent EIN where the Parent EIN does not exist in the ILM-EIN form.	Enter the correct Parent EIN.
Parent EIN, $Parent EIN$ is a component of Parent EIN $Temp Parent Parent$. Perform transaction on the Parent EIN, $Temp Parent Parent$ instead.	Attempting to perform an EIN transaction on a component.	Enter the correct Parent EIN.
One or more of these fields is not completed (Archive (P)arent-(C)omponent and Archive Type. Enter values into both of these fields and execute the transaction again.	Attempting to perform an archive transaction where the Archive (P)arent-(C)omponent and/or Archive Type fields is not completed.	Complete both of these fields.

Table 27.6.9-1. Non System-Failure Related Error Messages (3 of 5)

Error Message Char	Cause	Action
New Parent EIN is a component to Parent EIN ($Temp New Parent Parent$). Enter another New Parent EIN.	Attempting to relocate component(s) to a new Parent EIN where the new Parent EIN is a component to another EIN Structure.	Enter the correct new parent EIN.
New Parent EIN does not exist. Enter another New Parent EIN.	Attempting to relocate component(s) to a new parent EIN where the new parent EIN does not exist in the ILM-EIN form.	Enter the correct new parent EIN.
New Parent EIN ($New Parent EIN$) is not hardware. Enter another New Parent EIN value.	Attempting to relocate component(s) to a new parent EIN where the new parent EIN is something else other then hardware.	Enter the correct new parent EIN.
New Parent EIN ($New Parent EIN$) is the same as the old Parent EIN. Enter another new Parent EIN.	Attempting to relocate component(s) to a new parent EIN where the new parent EIN is the same as the old parent EIN	Enter the correct new parent EIN.
New Location values (New Location, New Building, or New Room) are not completed. Complete all the new locatioon values.	Attempting to perform a transaction where all the new location values are not completed.	Make sure all the new location values are completed.
Maintenance Work Order		
Parent EIN ($Parent EIN$) does not exist. Enter the correct Parent EIN.	Attempting to create a new maintenance work order where the Parent EIN does not exist in the ILM-EIN form.	Enter the correct Parent EIN.
EIN, $Parent EIN$, is not a parent. Enter the correct Parent EIN!	Attempting to create a new MWO where the parent EIN entered is a component of some EIN structure.	Enter the correct parent EIN.
ALDT 1 End Date-Time ($ALDT End Date-Time$) cannot be greater than current date.	Attempting to enter an ALDT End Date-Time that is greater than the current date and time.	Enter the correct ALDT End date-time.
Notification Date ($Notification Date-Time$) cannot be greater than current date.	Attempting to enter a notificcation Date-Time that is greater than the current date and time.	Enter the correct notification date-time.
Vendor Complete Date-Time ($Vendor Complete Date-Time$) cannot be greater than current date.	Attempting to enter a vendor complete Date-Time that is greater than the current date and time.	Enter the correct vendor complete date-time.
Vendor Call Date-Time ($Vendor Call Date-Time$) cannot be greater than current date.	Attempting to enter an vendor call Date-Time that is greater than the current date and time.	Enter the correct vendor call date-time.
ALDT Start Date-Time ($ALDT Start Date-Time$) cannot be greater than current date.	Attempting to enter an ALDT start Date-Time that is greater than the current date and time.	Enter the correct ALDT start date-time.

611-EMD-001

Table 27.6.9-1. Non System-Failure Related Error Messages (4 of 5)

Error Message Char	Cause	Action
Failure Date ($Failure Date-Time$) cannot be greater than current date.	Attempting to enter a failure Date-Time that is greater than the current date and time.	Enter the correct failure date-time.
Vendor Arrive Date-Time ($Vendor Arrive Date-Time$) cannot be greater than current date.	Attempting to enter a vendor arrive Date-Time that is greater than the current date and time.	Enter the correct vendor arrive date-time.
Vendor Initial Response Date-Time ($Vendor Initial Resp Date-Time$) cannot be greater than current date.	Attempting to enter a vendor initial response Date-Time that is greater than the current date and time.	Enter the correct vendor intial response date-time.
The MWO must already be created and you must have the MWO displayed in a Search/Modify window before clicking the Add Fail-Replacement Component Button!	Attempting to add a new work order line item where the work order information is blank.	Use the ILM-MWO form to find the appropriate work order and press the "Add Fail-Replacement Component" button to start adding line items to that work order.
Component EIN is the same as the MWO Parent EIN. Enter the correct component EIN.	Attempting to add a work order line item where the component EIN is the same as the MWO's Parent EIN.	Enter the correct component EIN.
New Parent EIN is the same as the MWO's Parent EIN. Enter the correct New Parent EIN.	Attempting to add a work order line item to move a component to a new EIN structure where the new parent EIN is the same as the MWO's Parent EIN.	Enter the correct new parent EIN value.
New Parent EIN ($New Parent EIN$) is a component to Parent EIN ($New Parent EIN Parent$). Enter the correct New Parent EIN value	Attempting to add a work order line item to move a component to a new EIN structure where the new parent EIN is the same as the MWO's Parent EIN.	Enter the correct new parent EIN value.
New Parent EIN does not exist. Enter the correct New Parent EIN.	Attempting to add a work order line item to move a component to a new EIN structure where the new parent EIN does not exist in the ILM-EIN form.	Enter the correct new parent EIN value.
New Parent EIN is not hardware. Enter the correct New Parent EIN value.	Attempting to add a work order line item to move a component to a new EIN structure where the new parent EIN in not hardware.	Enter the correct new parent EIN value.
Component EIN is a parent. Enter the correct Component EIN value.	Attempting to add a work order line item where the component EIN is a parent.	Enter the correct component EIN value.
Line Item does not have the correct event type and maint code. Enter the correct event type and maint code.	Entering the incorrect combination of event type and maint code in the ILM-MWO Line Item form.	Enter the correct event type and maint code.

Table 27.6.9-1. Non-System-Failure Related Error Messages (5 of 5)

Error Message Char	Cause	Action
License Management		
Entitlement Part No. does not exist. Enter the correct part number or add the new part into the ILM-License Entitlement Part form.	Attempting to associate entitlement with an Entitlement Part No. that does not exist in the ILM-License Entitlement Form.	Enter the correct part number or add the new part into the ILM-License Entitlement Part form.
Contract ID, $Contract ID$, already exists. Enter the correct Contract ID!	Attempting to associate a license entitlement with a contract ID where the contract id does not exist in the ILM-Maint contract form.	Enter the correct contract ID.
Entitlement ID does not exist. Enter the correct Entitlement ID.	Attempting to map a license to a purchased entitlement that does not exist in the ILM-Entitlement form.	Enter the correct Entitlement ID.
Node allocated is greater than Node remaining. Reduce number of Right-To-Use or enter another Entitlement ID.	Attempting to map a license to an entitlement where the entitlement node remaining is less than the amount allocating.	Reduce the number of node allocated.
User Allocated is greater than License Key User RTU allocated to the license. Enter the correct User Allocated value.	Attempting to map a license to an entitlement where the user allocated is greater than the license key user rights-to-use.	Reduce the user allocated to equal to or less than the license key RTU.
User Allocated is greater than User RTU Remaining. Lower User Allocated or Enter another Entitlement ID.	Attempting to map a license to an entitlement where the user allocated is greater than the entitlement user rights-to-use remaining	Reduce the user allocated.
This license right-to-use had already been mapped to entitlement $Ent ID Holder$.	Attempting to map a node lock license to more than one entitlement.	Do not map the license to another entitlement.
Total user allocated ($Total User Allocated$) is greater than the lic key user RTU. Reduce number of User Allocated	Attempting to map a license to entitlements where the total user allocated is greater than the license key user rights-to-use.	Reduce the user allocated to equal to or less than the license key RTU.
Node allocated or User allocated has not been assigned to this Entitlement ID ($Entitlement ID$).	Attempting to map a license to an entitlement where the user did not enter any value in the Node or user allocated.	Enter node or user allocated to map against the entitlement.
Node Allocated cannot be greater than one for nodelock licenses. Enter 1 to allocate 1 Right-To-Use for this Nodelock license.	Attempting to allocate more than 1 node rights-to-use for a node lock license.	Reduce the number of node rtu allocated to 1.
Combination of Entitlement Part No-MFR and Version $Temp PN_MFR_Ver$ already existed. Enter the correct Entitlement Part No.	Attempting to add a new entitlement part into the ILM-License Products form where the combination of the Entitlement Part No, MFR, and version already existed in the database.	Use the existing entitlement part information.

611-EMD-001

This page intentionally left blank.

28. Maintenance of Configuration Parameters

This chapter concerns the maintenance of configuration parameters that ECS servers access when they are started. The configuration parameters are manageable through a *Configuration Registry*. The Configuration Registry Server provides a single interface to retrieve configuration attribute-value pairs for ECS servers from the Configuration Registry Database, via a Sybase Server. The Configuration Registry Server maintains an internal representation of the tree in which configuration attribute-value pairs are stored. General configuration parameters used by many servers are stored in higher nodes in the tree. Parameters specific to a single ECS server are contained in the leaf nodes of the tree. ECS provides a script tool to load the Configuration Registry database from data in configuration files. This loading is a one-time event to populate the Registry database with the information contained in .CFG files. Once the Configuration Registry is loaded, if the configuration files are moved, renamed, or otherwise made inaccessible to the software, the software goes to the Configuration Registry to obtain needed configuration parameters. There is also a Configuration Registry GUI to view and edit configuration data in the database. Changes to the Configuration Registry are typically under the control of Configuration Management and the Database Administrator.

28.1 Parameter Change Control Procedure

Configuration parameters are 'controlled' by either DAAC or the EDF. Proposed changes to these configuration parameters originate from the controller of these parameters, in most cases. The one exception is when new ECS software/hardware patches or versions warrant new configuration parameters, or changes to the existing parameters. In cases where one of the entities, i.e., DAAC or EDF, proposes a change to an existing configuration parameter which it does not control, then the requesting entity follows the appropriate ECS change request, resolution and CCB approval process of the entity that controls that parameter. The ECS configuration change request process is described in Chapter 9.

Configuration Parameter Baseline documents define information on various areas of ECS. These areas include, but are not limited to the following:

- ECS Custom Code configuration, program and application configuration files and parameters.
- Sybase Server Databases.
- Operating System - build, options of auto mount, and Kernel configuration files and parameters
- COTS configuration files and parameters.

The information captured is site- and host-specific, wherever applicable. Whenever possible the following 'types' of information are captured:

- Configuration parameters and files.
- Definitions and descriptions of parameters.
- ECS recommended value.

- Value or value ranges; i.e., common range across multiple DAACs or DAAC-specific range.
- Impacts associated with changing a parameter.
- Controlling entity of the parameter (i.e., DAAC or EDF).

The baselined documents are maintained by ECS CM, and are posted on the ECS Baseline Information System (EBIS) Site. The baselined information is replicated and posted on a mirror site for DAACs' accessibility and review.

All changes to ECS software/hardware patches and versions are controlled by the relevant CCBs. (Refer to Chapter 9.)

A prototype (non-baselined) version of the configuration information may be posted on the EBIS and mirror site for ECS and DAAC review prior to the CCB approval process, as long as the posted information is clearly identified as 'prototype' to distinguish it from the ECS baselined information.

28.2 Overview of Configuration Parameter Files

Various types of source files are used to modify the configuration parameters at the Landover functionality labs and at the DAACs. These are as follows:

- The 'Mkcfg' scripts contain persistent values of configuration parameters, and only ECS developers are allowed to modify them.
- The '.cfgparms', '.extparms' and '.dbparms' files, which are resident in Clearcase and contain persistent values of configuration parameters for the EDF. The delivered version of these files to the DAACs contain DAAC specific and mode specific values. The .cfgparms files hold parameters used to generate the standard .CFG files. The .extparms files contain parameters to generate non-standard (or extension)files, and are not used by most subsystems. The .dbparms files hold parameters used in database operations.
- The .cfgparms file patches the .cfgparms, .extparms and .dbparms. Changes are from one version to another, e.g., 6A.01 to 6A.02.
- The '.rgypatch' file patches the registry database.

28.3 Overview of Configuration Registry

The Registry Database is initially created with parameters from the .CFG files that are generated by running ECS Assistant's mkcfg process. In future other files such as the .PCFG and .ACFG files may be housed in the registry database. A .rgypatch file is required to make a change (i.e., to add, update or delete a parameter) to a registry configuration parameter. Another method of making changes to the parameters in the registry is to use the Registry GUI.

ECS servers use a single class to access. CFG files. This class was modified to request values from the registry servers, if there is no .CFG file present. The registry server then obtains the requested values from the registry database, and returns them to the requester. It is important to note that the registry server provides read-only access to the configuration parameters, and caches non-leaf nodes. Therefore changes made via the GUI which is mode specific, do not take

effect until the affected servers have been restarted. If a non-leaf node is added, deleted, or updated, the changes do not take affect until the registry server has been restarted and then the affected servers have been restarted.

The registry schema contains provision for Access Control List (ACL) processing. This restricts read and write privileges. For example, the write privileges of the registry in the OPS mode could be granted to individuals in a supervisory role, while read privileges could be non-supervisory.

28.3.1 Registry Deployment and Baseline Maintenance

Registry changes can be made by anyone having the DBO privileges. Any changes to the configuration parameter baseline, such as addition, deletion or modifications of parameters, should follow the respective CCB process.

ECS servers use the .CFG files, if present in /usr/ecs/<mode>/CUSTOM/cfg directory when the server is started. Otherwise the ECS servers use the registry database for configuration parameters. Therefore, moving or renaming the .CFG files causes servers to use the registry the next time they restart. This allows the registry to be brought online, and taken offline on a server-by-server basis.

Maintenance of parameters in the registry must be via the registry GUI or via database patches. Changes to the .CFG files or the .cfgparms files will not result in changes to the registry unless the files are re-imported into the registry. The ability to create .CFG files during the "make config" phase of ECS Assist processing can be switched off (via a configuration parameter). The addition, deletion and modification of values in the registry are achieved either by the GUI or by a database patch script.

The EDF maintains the "baseline" registry database that is used to generate the database patches. The baseline registry database has the structure of a generic DAAC, using functional host names rather than actual host names. It contains the master values of parameters owned by the EDF, and contains null values for parameters owned by the configuration management process. ECS developers make changes to the baseline database via the software configuration management process. The database contains an attribute tree for each release, and release patch supported by the EDF. Prior to a release or a release patch, a configuration patch script is created by comparing the tree for the new release with the tree for the release being replaced.   The patch script contains a series of add, update and delete statements, tagged to indicate the sites they pertain to. The primary purpose of the patch is to enhance the process of parameter additions, modifications or deletions to the DAAC configuration. The patch is also used to propagate value changes for parameters owned by the EDF.

28.4 Configuration Registry Procedures

28.4.1 Registry Preparation Procedure

Create a new Registry Tree for the mode to be installed:

1   Ensure that the registry database has been backed up.
- For detailed instructions refer to Section 28.4.2, **Registry Database Backup Procedure**.

2   Start the registry GUI for the mode into which the required Drop is being installed.

3   Select the attribute tree name that is mapped to the current mode from the drop-down menu
- When the correct tree is selected, the name of the mode should appear in the "**Attribute Information**" box below).

4   Click on the name of the attribute tree in the window below (the window with a white background), to enable the menu icons.

5   Click on the **Copy selected item** icon (the icon on the left).

6   Enter the name of the new registry tree in the **Attribute Name** box.

7   Select the **Ok** button.

8   Wait for the new tree to be created.

NOTE: It may take a long time for the new tree to be created -- on the order of 30 minutes).

9   Select the newly-created tree name in the window.

10   Click on the **MAP** icon.

11   Select the mode name from the drop-down menu in the new window that appears.

12   Select **Ok**.

13   Exit the GUI by selecting the **File → Exit** menu option.

28.4.2   Registry Database Backup Procedure

The Registry database should be regularly (e.g., nightly) backed up via the Sybase dumps. (The DAACs should have added the registry db to their list of dbs for backup).

1   To make a backup copy of a tree within the registry, just click on the "copy" in the registry GUI before making modifications to the tree.

2   To make a backup copy of a tree and dump it to a file first click on **Add New Tree** at the bottom of the tree display in the GUI.

3   Select the **MkRgyPatch** option from the menu

4   Specify the new tree (which is empty) and the tree that you want to copy.
- This produces a file containing the tree in the rgypatch format.

28.4.3 Registry Patch Procedure

Patch the Registry Database using the **.rgypatch** file:

1   From the ECS Assist Subsystem Manager, select the appropriate Mode, Subsystem, and Component from the main window.

2   Select **Registry Data Patch** from the **Tools** menu.
- An **Apply Registry Data Patch** window is displayed.

3   In the **Apply Registry Data Patch** window enter the name of the SQL server in the **Registry Database Server:** box.

4   Enter the registry database DBO ID.

5   Enter the registry database password.

6   In the **Registry DB Name:** box enter the name of the registry database (**EcCsRegistry_<mode>**) and press the **ENTER** key.
- ECS Assist connects to the registry database and populates the drop-down menu associated with the next field (**Tree to patch:**).

7   Use the drop-down menu to select the appropriate registry tree that is being patched.
- If unsure which tree to select, bring up the registry GUI and verify which tree is mapped to the mode being updated.

8   Click on the **Select Patch File** button to bring up the **File Selection Dialog** window.

9   Navigate through this window to find the .rgypatch file.
- If the installation was successful, it should appear in the /usr/ecs/<MODE>/CUSTOM/.installed/DMS directory).

10   Highlight the .rgypatch file in the window and select **OK**.

11   Verify that the appropriate information is indicated in the **Patch File:** box in the **Apply Registry Data Patch** window and select **OK**.

- At this point, the registry patch will be applied.

12   Monitor the output via ECS Assist for any warning or error messages as the patch is run.

28.4.4   Display Parameters Using the Configuration Registry GUI

Display parameters using the Configuration Registry GUI:

1   On workstation **x0dms##**, at the UNIX prompt in a terminal window, type **/usr/ecs/*mode*/CUSTOM/utilities/EcCsRegistryGUIStart *mode* &** at a UNIX command prompt and then press the **Return/Enter** key (where *mode* is likely to be **TS1**, **TS2**, or **OPS**).

- NOTE: The **x** in the workstation name will be a letter designating your site: **g** = GSFC, **m** = SMC, **l** = LaRC, **e** = EDC, **n** = NSIDC, **o** = ORNL, **a** = ASF, **j** = JPL; the **##** will be an identifying two-digit number (e.g., **g0dms03** indicates a data management subsystem workstation at GSFC). If you access the workstation through a secure shell remote login (ssh), you must enter **setenv DISPLAY <local_workstation IP address>:0.0** prior to the ssh before entering the command after the ssh. The **<ipaddress>** is the ip address of **x0mss##**, and **xterm** is required when entering this command on a Sun terminal.
- The Database Login window is displayed with entries filled in for **User Id:** (e.g., **EcCsRegistry**), **Server:** (e.g., **x0icg02_srvr**), and **DB Name:** (e.g., **EcCsRegistry_mode**).

2   In the Database Login window, click in the **Password:** field and type the password.

- The typed password is not displayed (dots are displayed in place of the password).

3   Click on the **Sign On** button.

- The Database Login window is closed and the Configuration Registry GUI is displayed.

4   On the tree showing system hosts displayed on the left side of the GUI, click on the "+" sign next to one of the hosts for which parameters are to be displayed.

- The tree displays a **config** branch.

5   Click on the "+" next to **config**.

- The tree displays a **CFG** branch.

6   Click on the "+" next to **CFG**.

- The tree displays the computer software components for the selected host.

7   Click on one of the listed components (or its folder icon).

- The **Attribute Listing** field displays the configuration parameters associated with the selected component. If there are a large number of parameters, the right side of the window will have a scroll bar that may be used to scroll down the list.

8   Click on one of the listed parameters.

- The **Attribute Information** pop-up window for the selected parameter is displayed, showing detailed information concerning the parameter.
- If you are logged in with an account authorized with appropriate permissions, the **Attribute Information** window permits changing or deleting the parameter.

9   To exit from the Configuration Registry GUI, follow menu path **File→Exit**.

This page intentionally left blank.

Appendix A. Additional Material

Examples of the Various ODL Files Used by Each Instrument Team

Section 26.13 deals, in part, with the use of ODL files in SSI&T activities. This Appendix serves as a supplement and reference for that section. Useful examples of ODL files follow. ODL Template files, from which specific examples were created, are listed first. Then, examples of specific ODL files are listed by instrument (ASTER, MISR, MODIS and AIRS). *Please note that in many of the examples that follow, much of the instrument/ECS provided comments have been deleted in order to keep this document reasonably short.*

A.1   Template ODL Files

There are five Template ODL files listed here. The specific or tailored ODL files listed in Sections A.2 through A.5 were derived from these templates by appropriate editing and filling-in of values (*NOTE: while the TILE ODL file is currently not being used by any of the instrument teams mentioned above, the template is included here for completeness*). The five ODL Template files listed reside, on the AIT Sun host, at /usr/ecs/<mode>/CUSTOM/data/DPS. They are:

PGE_ODL.template
ESDT_ODL.template
ORBIT_ODL.template
TILE_ODL.template*
PATHMAP_ODL.template

A.1.1 PGE_ODL.template

```
/*********************************************************************/
/*                                                   */
/*         TEMPLATE PGE SCIENCE METADATA ODL FILE              */
/*                                                   */
/*                                                   */
/* The SSIT operator's responsibility is to copy this file over and    */
/* edit it to add all necessary PDPS metadata values.            */
/*                                                   */
/* All PGE ODL files must reside in directory $DPAT_PGE_SCIENCE_MD.    */
/* This directory is now set through the Process Framework CFG files.  */
/*                                                   */
/* The operator must add a value to the right of the "=" for each     */
/* parameter.                                         */
/*                                                   */
/* Normally, a template version of this file (without the comments)   */
/* will be generated by the SSIT operator from the PCF delivered to    */
/* SSIT. This file is meant to show the SSIT personnel and the        */
/* Instrument teams the information that is needed for a PGE to be     */
/* planned and executed by the Planning and Data Processing system of  */
/* ECS.                                              */
/*                                                   */
```

```
/* CHANGE LOG                                                         */
/*     -- Added new schedule type for Data Scheduled PGEs.   02/18/98    */
/*            Changed QUERY_DELAY to be optional for all PGEs.         */
/*            Changed SPATIAL_KEY_INPUT to KEY_INPUT                   */
/*     -- Fixed description for Begin/End Period Offsets.   03/10/98    */
/*     -- Added The Distinct Value definition.             03/26/98    */
/*     -- Fixed length of CATEGORY.                       03/27/98    */
/*     -- Fixed length of FILETYPE_NAME.                   03/31/98    */
/*     -- Added PATHMAP_NAME                              04/13/98    */
/*     -- Changed how WAITFOR is suposed to be set.        05/06/98    */
/*            Added entries for ASSOCIATED_SCIENCE_DATA to handle  */
/*            BROWSE and QA products.                                 */
/*     -- Added START_OF_MINUTE to PROCESSING_BOUNDARY.    06/24/98    */
/*            Updated DATA DAY values for PGE_PARAMETER_DYNAMIC_VALUE, */
/*            and DATABASE_QUERY                                      */
/*     -- Added KEY_PARAMATER_NAME and KEY_PARAMETER_VALUE  07/05/98    */
/*            for Metadata Checks and Metadata Queries.               */
/*     -- Updated description for KEY_PARAMETER_NAME and    07/11/98    */
/*            KEY_PARAMETER_VALUE.                                    */
/*     -- Updated lengths for PLATFORM and INSTRUMENT.      08/13/98    */
/*     -- Updated explination for "Already Created Tile"    08/18/98    */
/*            for QUERY_TYPE.                                         */
/*     -- Added CHECK_FOR_OUTPUT flag.                     08/24/98    */
/*     -- Added MOST_RECENT_QUERY_OFFSET and MOST_RECENT_  09/02/98    */
/*            QUERY_RETRIES parameters for the Most Recent Granule */
/*            Production Rule.                                        */
/*     -- Added AUXILIARY_LOGICAL_ID object for handling_   09/23/98    */
/*            multiple L0 granules.                                   */
/*        Removed older change commentary.                            */
/*     -- Added COMPOUND_PGE parameter for handling        10/23/98    */
/*            PGEs with multiple executables.  Also deleted           */
/*            old change history.                                     */
/*     -- Updated description for ALTERNATE INPUT TIMER     11/07/98    */
/*            to say that it has not affect for Dynamic               */
/*            Internal ESDTs.                                         */
/*     -- Added ALIGN_DPR_TIME_WITH_INPUT_TIME parameter    12/20/98    */
/*     -- Increased the number of Profile Ids from 99 to    07/12/99    */
/*            999.                                                    */
/*        Removed restriction on ALTERNATE_INPUT_TIMER with respect    */
/*            to Internal Dynamic ESDTs.                              */
/*        Added PGE_DEFAULT_PROFILE  parameter.                       */
/*        Added PROFILE_SELECTOR_PGE_PARAMETER.                       */
/*      -- Added Closest Granule values:                   08/19/99    */
/*        CLOSEST_QUERY_OFFSET, CLOSEST_QUERY_RETRIES, and            */
/*        CLOSEST_QUERY_DIRECTION.                                    */
/*      -- Added "Metadata" to the query type.            12/13/99    */
/*      -- Updated Toolkit logical Ids that SSIT allows      02/03/00    */
/*        in ODL.                                                    */
/*        Removed old change history                                 */
/*                                                                    */
/*********************************************************************/


/*********************************************************************/
/*          PGE name                                                  */
/*             -- Must be a string, max len 10 characters            */
/*             -- PGE name inside ODL file must be identical to       */
/*                    PGE name used as part of ODL filename           */
/*          Example                                                   */
```

```
/*              PGE_NAME = "ssit"                                        */
/**********************************************************************/

PGE_NAME = ""

/**********************************************************************/
/*          PGE version                                         */
/*              -- Must be a string, max len 5 characters       */
/*              -- PGE version inside ODL file must be identical to    */
/*                  PGE version used as part of ODL filename        */
/*          Example                                             */
/*              PGE_VERSION = "1.0"                                  */
/**********************************************************************/

PGE_VERSION = ""

/**********************************************************************/
/*          PGE Profile ID                                      */
/*              -- Must be an integer                                  */
/*              -- Must be >= 0 and <= 999                      */
/*          Example                                             */
/*              PROFILE_ID = 99                                      */
/**********************************************************************/

PROFILE_ID =

/**********************************************************************/
/*          PGE Profile Description                             */
/*              -- Must be a string, max length 255 characters       */
/*          Example                                             */
/*          PROFILE_DESCRIPTION = "Improved performance numbers"      */
/**********************************************************************/

PROFILE_DESCRIPTION = ""

/**********************************************************************/
/*          PGE On-Demand Profile Default                       */
/*              -- Must be a string, set to "Y" or "N".              */
/*              -- If NOT Present, defaults to "N".             */
/*            -- Marks a particular for this PGE (PGE Name +          */
/*                PGE version) as the default for On Demand Processing */
/*                Requests.                                     */
/*              -- If more than 1 PGE (PGE Name + PGE Version) has this */
/*                value set, an error will be returned.              */
/*          Example                                             */
/*              PGE_DEFAULT_PROFILE = "N"                            */
/**********************************************************************/

PGE_DEFAULT_PROFILE = ""

/**********************************************************************/
/*          Spacecraft platform name                            */
/*              -- Must be a string, max len 25 characters      */
/*          Example                                             */
/*              PLATFORM = "TRMM"                               */
/**********************************************************************/

PLATFORM = ""
```

```
/*******************************************************************/
/*        Instrument name                                  */
/*            -- Must be a string, max len 20 characters        */
/*        Example                                     */
/*            INSTRUMENT = "CERES"                          */
/*******************************************************************/

INSTRUMENT = ""

/*******************************************************************/
/*        Minimum Number of Outputs                        */
/*          (used for QA purposes)                         */
/*            -- Must be a integer, maxium 3 digits.           */
/*        Example                                     */
/*            MINIMUM_OUTPUTS = 0                           */
/*******************************************************************/

MINIMUM_OUTPUTS =

/*******************************************************************/
/*        Type of PGE Scheduling                           */
/*            -- Must be a string with one of the following values:   */
/*            "Time" = TimeScheduled (PGE is scheduled based on the */
/*                  boundary/period and the arrival of data).     */
/*            "Data" = DataScheduled (PGE is scheduled based on the */
/*                  avialability of data produced by other       */
/*              PGEs).                             */
/*            "Tile" = TileScheduled (PGE is scheduled based on the */
/*                  the definition of Tiles).  Note that        */
/*                  TILE_SCHEME_NAME must have a value for Tile   */
/*                  Scheduled PGEs.                        */
/*            "Orbit" = OrbitScheduled (PGE is scheduled based     */
/*                   the orbit of the spacecraft. Note that then */
/*                PROCESSING_PERIOD must = "ORBITS=1" and      */
/*                   PROCESSING_BOUNDARY must =                */
/*                   "START_OF_ORBIT".  Also,  A file of named   */
/*                ORBIT_<platform>.odl must be present.       */
/*               Also if you want a Pathmap it needs to be    */
/*               specified under PATHMAP_NAME.           */
/*            "Snapshot" = SnapshotScheduled (PGE is scheduled      */
/*                  based on a single date/time entered       */
/*                  entered when the production request is    */
/*                  submitted.                        */
/*        Example                                     */
/*            SCHEDULE_TYPE = "Tile"                               */
/*******************************************************************/

SCHEDULE_TYPE = ""

/*******************************************************************/
/*        Nominal time interval between start of PGE runs        */
/*            -- NOT needed for PGEs where SCHEDULE_TYPE = "Snapshot"  */
/*          or SCHEDULE_TYPE = "Data".                       */
/*            -- Must contain a single P=V string, where          */
/*            P is one of { YEARS, MONTHS, THIRDS WEEKS, DAYS,     */
/*                    HOURS, MINS, SECS, ORBITS}               */
/*            -- NOTE that ORBITS must be used for PGEs based on an    */
/*            Orbit Model.  Note that PROCESSING_BOUNDARY must be   */
/*            set to "START_OF_ORBIT".                       */
```

```
/*          Example                                                    */
/*              PROCESSING_PERIOD = "DAYS=1"                       */
/********************************************************************/


PROCESSING_PERIOD = ""


/********************************************************************/
/*          Nominal time boundary on which PGE processing begins   */
/*              -- NOT needed for PGEs where SCHEDULE_TYPE = "Snapshot"  */
/*              or SCHEDULE_TYPE = "Data".                         */
/*               -- Must contain a one of                          */
/*                 { START_OF_MINUTE, START_OF_HOUR, START_OF_6HOUR,   */
/*                   START_OF_DAY, START_OF_WEEK,                 */
/*                   START_OF_ONE_THIRD_MONTH,                    */
/*                   START_OF_MONTH, START_OF_YEAR, START_DATE,      */
/*                   START_OF_ORBIT };                            */
/*                also, "+<n>" or "-<n>"may be added to any of these,  */
/*                where <n> specifies integer seconds.            */
/*                For START_DATE an "=" can be added followed by the  */
/*                 start date.                                    */
/*              -- NOTE that START_OF_ORBIT must be used for PGEs based  */
/*                on an Orbit Model.  A file of named            */
/*                ORBIT_<platform>.odl must be present.             */
/*          Example                                               */
/*              PROCESSING_BOUNDARY = "START_OF_HOUR"              */
/********************************************************************/


PROCESSING_BOUNDARY = ""


/********************************************************************/
/*          Software version                                      */
/*              -- Must be a string, max 5 len characters         */
/*              -- If Ssw version is not the same as PGE version,   */
/*                  SswId ("<PGE Name>#<Ssw Version>") must already   */
/*                  be defined in the database;                   */
/*                  That is, the only allowed values of the         */
/*                  software version are either this PGE version  */
/*                  or a previous PGE version for this PGE name    */
/*          Example                                               */
/*              PGE_SSW_VERSION = "1.0"                           */
/********************************************************************/


PGE_SSW_VERSION = ""


/********************************************************************/
/*          Delay for query                                       */
/*               -- Optional for types of PGEs.                   */
/*             -- The amount of time (in SECONDS) that the query for   */
/*                 input data should be delayed. This value is added */
/*              onto the Stop Time of any DPR generated with this    */
/*              PGE.                                              */
/*              -- Used for Tiling or Metadata Query inputs.        */
/*              -- OPTIONAL Parameter.  If not specified it is set to 0.  */
/*              -- Must be an integer value >= 0.                 */
/*          Example                                               */
/*              QUERY_DELAY = 360 (1 hour)                        */
/*                                                                */
/********************************************************************/
```

```
QUERY_DELAY = 0

/**********************************************************************/
/*          Name of the Tiling Scheme used                          */
/*              -- Must be a string of at most 20 characters.       */
/*            -- There can be NO spaces in the string.              */
/*              -- A file that defines the Tiling Scheme must       */
/*                 be created with the name TILE_<tiling scheme>.odl */
/*           Example                                                 */
/*              TILE_SCHEME_NAME = "Earth_Squared"                  */
/*                                                          */
/* NOTE that this is only needed for PGEs of Schedule Type = "Tile". */
/* It can be deleted for all other types of PGEs.                    */
/**********************************************************************/

TILE_SCHEME_NAME = ""

/**********************************************************************/
/*          Name of Pathmap used                                    */
/*              -- Must be a string of at most 25 characters.       */
/*            -- There can be NO spaces in the string.              */
/*              -- A file that defines the Pathmap must             */
/*                 be created with the name PATHMAP_<Pathmap_Name>.odl */
/*           Example                                                 */
/*              PATHMAP_NAME = "Some_Name"                          */
/*                                                          */
/* NOTE that this is only needed for PGEs of Schedule Type = "Orbit". */
/* It can be deleted for all other types of PGEs.                    */
/**********************************************************************/

PATHMAP_NAME = ""

/**********************************************************************/
/* OPTIONAL PARAMETER                                                */
/*          Check For Outputs                                        */
/*              -- Must be  a character value of either "Y" (YES)    */
/*                 or "N" (NO).                                      */
/*              -- Defaults to "N" if not specified.                */
/*              -- When set to "Y", this means that a DPR of the PGE */
/*                 will ONLY be scheduled if the output of that PGE has */
/*                 NOT been produced.  This is currently planned for use */
/*                 in ASTER Routine Processing.                      */
/*              -- Note that creating a DPR (in the Production Request */
/*                 Editor) with Reprocessing set will override this  */
/*                 flag.                                             */
/*           Example                                                 */
/*              CHECK_FOR_OUTPUTS = "N"                             */
/**********************************************************************/

CHECK_FOR_OUTPUTS = "N"

/**********************************************************************/
/* OPTIONAL PARAMETER                                                */
/*          Compound Pge Flag                                        */
/*              -- Must be a character value of either "Y" (YES)     */
/*                 or "N" (NO).                                      */
/*              -- Defaults to "N" (Not Compound PGE) if not specified. */
/*              -- When set to "Y", this means that this PGE is made up */
/*                 of multiple executables AND that the output of one */
```

```
/*             of these executables is the input of another          */
/*             executable within the PGE.                       */
/*           -- Note that setting this flag will hurt the performance   */
/*             of the Destaging step during PGE execution.  It is      */
/*             best to only set it to "Y" if both conditions          */
/*             mentioned above are true.                      */
/*         Example                                  */
/*             COMPOUND_PGE = "N"                           */
/**********************************************************************/

COMPOUND_PGE = "N"

/**********************************************************************/
/* Exit message object                                   */
/*                                           */
/* Defines a possible PGE exit code, and associates a message with it.  */
/*                                           */
/* This object is optional and can be deleted if no EXIT MESSAGEs are   */
/* desired.                                       */
/*                                           */
/* Replicate the object as needed to define EXIT MESSAGEs for multiple  */
/* EXIT CODEs.                                     */
/*                                           */
/* See "Establishing Science Software Exit Conditions for the        */
/* Production Environment" white paper (420-WP-006-002) for the      */
/* definitions and of exit code values and their uses.            */
/*                                           */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present        */
/**********************************************************************/

OBJECT = EXIT_MESSAGE

/**********************************************************************/
/*         Class (object counter, used only to distinguish objects)   */
/*             -- Must be an integer                      */
/*             -- Must be unique in this file for this type of object   */
/*             -- Must be greater than 0.                    */
/*         Example                                */
/*             CLASS = 1                               */
/**********************************************************************/

   CLASS= 1

/**********************************************************************/
/*         Exit code for this PGE                        */
/*             -- Must be an integer                      */
/*             -- Must be 0 or between 200 and 239            */
/*         Example                                */
/*             EXIT_CODE = 200                            */
/**********************************************************************/

   EXIT_CODE = 0

/**********************************************************************/
/*         Message corresponding to this exit code             */
/*              -- Must be a string, max len 240 characters      */
/*         Example                                */
/*             EXIT_MESSAGE = "PGE successfully completed"        */
/**********************************************************************/
```

```
      EXIT_MESSAGE = ""

/**********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/**********************************************************************/

END_OBJECT = EXIT_MESSAGE

/**********************************************************************/
/* Exit dependency object                                    */
/*                                              */
/* Defines names, exit codes and conditions of PGEs on which this */
/* PGE depends.                                     */
/*                                              */
/* This object is optional and can be deleted if no EXIT DEPENDANCY(s)  */
/* exist for this PGE.                                */
/*                                              */
/* Replicate this object as needed to define multiple EXIT        */
/* DEPENDANCies for the PGE.                            */
/*                                              */
/* See "Establishing Science Software Exit Conditions for the      */
/* Production Environment" white paper (420-WP-006-002) for the   */
/* definitions and of exit code values and their uses.          */
/*                                              */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/**********************************************************************/

OBJECT = EXIT_DEPENDENCY

/**********************************************************************/
/*         Class (object counter, used only to distinguish objects)   */
/*             -- Must be an integer                        */
/*             -- Must be unique in this file for this type of object  */
/*             -- Must be greater than 0.                     */
/*         Example                                 */
/*             CLASS = 1                              */
/**********************************************************************/

   CLASS= 1

/**********************************************************************/
/*         Name of PGE upon which this PGE depends             */
/*             -- Must be a string, max len 10 characters       */
/*             -- SswId ("<PGE name>#<Ssw version>") must be different  */
/*                than  this SswID (PGE cannot depend on itself)  */
/*             -- SswId must already exist in the database       */
/*         Example: This CERES PGE depends on the exit code of       */
/*                a MODIS PGE: execute the CERES PGE only if the     */
/*                MODIS PGE had exit code = 0                */
/*             DEPENDENCY_PGE_NAME = "MODIS"                    */
/**********************************************************************/

   DEPENDENCY_PGE_NAME = ""

/**********************************************************************/
/*         Version of Ssw upon which this Ssw depends             */
/*             -- Must be a string, max len 5 characters       */
/*             -- SswId ("<PGE name>#<Ssw version>") must be different  */
```

```
/*              than  this SswID (PGE cannot depend on itself)  */
/*         Example                                              */
/*            DEPENDENCY_SSW_VERSION = "x"                      */
/************************************************************************/

   DEPENDENCY_SSW_VERSION = ""

/************************************************************************/
/*        Operator for exit code dependency condition            */
/*            -- Must be one of { >, <, >=, <=, =, != }       */
/*         Example                                           */
/*            EXIT_OPERATION = "="                           */
/************************************************************************/

   EXIT_OPERATION = ""

/************************************************************************/
/*        Exit code for PGE upon which this PGE depends          */
/*            -- Must be an integer                          */
/*            -- Must be 0 or between 200 and 239            */
/*            -- Must already exist in the database as a valid   */
/*                exit code for the PGE upon which this PGE depends   */
/*         Example                                           */
/*            EXIT_CODE = 0                                  */
/************************************************************************/

   EXIT_CODE = 0

/************************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present      */
/************************************************************************/

END_OBJECT = EXIT_DEPENDENCY

/************************************************************************/
/* PCF entry object                                          */
/*                                                        */
/* The program DpAtCreateOdlTemplate (run at SSIT) generates one of   */
/* these object for each file entry in the PCF.  Only generic Toolkit  */
/* Logical IDs are ignored during Template Creation.              */
/*                                                        */
/* The operator needs to fill in values for the parameters as described */
/* in the comments for each parameter. Note that some parameters   */
/* must be filled for each PCF entry, while others are optional or only */
/* needed based on the values of other parameters.               */
/*                                                        */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED                    */
/************************************************************************/

OBJECT = PCF_ENTRY

/************************************************************************/
/*        Class (object counter, used only to distinguish objects)   */
/*            -- This line is generated by DpAtCreateOdlTemplate */
/*                from the PCF and is normally not modified      */
/*            -- Must be an integer                          */
/*            -- Must be unique in this file for this type of object   */
/*            -- Must be greater than 0.                      */
/*         Example                                           */
```

```
/*              CLASS = 1                                             */
/**********************************************************************/

   CLASS = 1


/**********************************************************************/
/*          PCF logical ID                                    */
/*              -- This line is generated by DpAtCreateOdlTemplate */
/*                 from the PCF and is normally not modified.        */
/*              -- Must be a positive integer.                       */
/*              -- Most values between 10000 and 10999 (Toolkit specific */
/*                 Logical IDs) are ignored except for the following:   */
/*                 Data Dictionary Logical ID (10251)        */
/*                 Attitude Data Logical ID (10501)          */
/*                 Ephmerous Data Logical ID (10502)         */
/*                 Math Constant Logical ID (10999)          */
/*                     Index Data File Logical ID (10900)        */
/*                     DEM Logical Ids (10649 - 10655)               */
/*                 Ascii Dump Logical ID (10255)             */
/*                     Disable Status Level RTI Logical ID (10117)    */
/*                 Disable Seed RTI Logical ID (10118)        */
/*                     Disable Status code RIT Logical ID (10119)     */
/*          Example                                           */
/*              LOGICAL_ID = 100                                     */
/**********************************************************************/

   LOGICAL_ID = 100


/**********************************************************************/
/*          PCF file type                                     */
/*              -- This line is generated by DpAtCreateOdlTemplate */
/*                 from the PCF and is normally not modified       */
/*              -- Must be an integer between 1 and 8 inclusive       */
/*              =1, PRODUCT INPUT FILES                       */
/*              =2, PRODUCT OUTPUT FILES                      */
/*              =3, SUPPORT INPUT FILES                       */
/*              =4, SUPPORT OUTPUT FILES                      */
/*              =5, USER DEFINED RUNTIME PARAMETERS              */
/*              =6, INTERIM/INTERMEDIATE INPUT FILES            */
/*              =7, INTERIM/INTERMEDIATE OUTPUT FILES                */
/*              =8, TEMPORARY I/O                             */
/*          Example                                           */
/*              PCF_FILE_TYPE = 1                                    */
/**********************************************************************/

   PCF_FILE_TYPE = 1


/**********************************************************************/
/*          Data Type Name -- same as Data Server ESDT Short Name  */
/*              -- Must be a string, max len 8 characters        */
/*              -- Required for all PCF ENTRY objects, except those with */
/*                 PCF_FILE_TYPE = 5 or 8                     */
/*              -- An ESDT ODL file for this name must exist in      */
/*                 in directory $DPAT_ESDT_SCIENCE_MD, and have a name   */
/*                 of the form                               */
/*                  "ESDT_<Data Type Name#Data Type Version>.odl"  */
/*              -- An ESDT of this Short Name must already be defined    */
/*                 at the Data Server                              */
/*          Example                                           */
```

```
/*              DATA_TYPE_NAME = "TRWpl182"                    */
/*         implies file $DPAT_ESDT_SCIENCE_MD/ESDT_TRWpcal182.odl */
/*         already exists in the SSIT environment, and that      */
/*         ESDT Short Name "TRWpl182" already exists in the      */
/*         Data Server                                           */
/****************************************************************/

   DATA_TYPE_NAME = ""

/****************************************************************/
/*         Data Type Version                                    */
/*            -- Must be a string, max len 5 characters         */
/*            -- Required for all PCF ENTRY objects, except those with */
/*               PCF_FILE_TYPE = 5 or 8                         */
/*            -- An ESDT ODL file for this name must exist in    */
/*               in directory $DPAT_ESDT_SCIENCE_MD, and have a name   */
/*               of the form                                     */
/*                "ESDT_<Data Type Name#Data Type Version>.odl"  */
/*            -- An ESDT of this Short Name and Version must already   */
/*               be defined at the Data Server                  */
/*         Example                                               */
/*            DATA_TYPE_VERSION  = "3.5.1"                       */
/****************************************************************/

   DATA_TYPE_VERSION = ""

/****************************************************************/
/*         Minimum number of input granules for this logical ID */
/*            -- This line is generated by DpAtCreateOdlTemplate */
/*               from the PCF and is only modified if the PGE can */
/*            execute successfully with fewer granules than in the */
/*            PCF from which the template was generated.        */
/*            -- Used to support "Minimum Number of Granules"    */
/*            Production Rule.                                   */
/*             -- Required for all PCF ENTRY objects            */
/*               PCF_FILE_TYPE = 1, 3, 6 (ignored otherwise).         */
/*             -- Must be a >= 0.                               */
/*            -- Note that for number of files within a granule  */
/*            greater than one, the FILE TYPE object for this entry */
/*            must be changed to specify the various file types and */
/*            maximum number of files.                          */
/*         Example                                               */
/*            MIN_GRANULES_REQUIRED = 1                          */
/****************************************************************/

   MIN_GRANULES_REQUIRED = 1

/****************************************************************/
/*         Maximum number of input granules for this logical ID */
/*             -- This line is generated by DpAtCreateOdlTemplate */
/*               from the PCF and is only modified if the PGE can */
/*            execute successfully with more granules than in the */
/*            PCF from which the template was generated.        */
/*            -- Used to support "Minimum Number of Granules"    */
/*            Production Rule.                                   */
/*             -- Required for all PCF ENTRY objects            */
/*               PCF_FILE_TYPE = 1, 3, 6 (ignored otherwise)          */
/*             -- Must be a positive integer                    */
/*            -- Note that for number of files within a granule  */
```

```
/*              greater than one, the FILE TYPE object for this entry   */
/*              must be changed to specify the various file types and    */
/*              maximum number of files.                                 */
/*         Example                                                       */
/*              MAX_GRANULES_REQUIRED = 1                                */
/***********************************************************************/

     MAX_GRANULES_REQUIRED = 1

/***********************************************************************/
/*         Begin Period Offset.                                         */
/*              -- Only needed if data for this PCF entry is to be       */
/*                 selected BEFORE (-) or AFTER (+) the period defined   */
/*                 for the ESDT (stated in the corresponding ESDT        */
/*                 ODL file).                                            */
/*              -- Defaulted to 0.                                       */
/*              -- If set, must be an integer number of seconds.         */
/*                 A positive value indicates that the value is BEFORE   */
/*                 the Period of the ESDT.  A Negative value is added to */
/*                 the Period so that the data will be found after the   */
/*                 start of the period specified for the ESDT.           */
/*         Example                                                       */
/*              BEGIN_PERIOD_OFFSET = "7200" (2 hours)                   */
/***********************************************************************/

     BEGIN_PERIOD_OFFSET = 0

/***********************************************************************/
/*         End Period Offset.                                           */
/*              -- Only needed if data for this PCF entry is to be       */
/*                 selected BEFORE (-) or AFTER (+) the period defined   */
/*                 for the ESDT (stated in the corresponding ESDT        */
/*                 ODL file).                                            */
/*              -- Defaulted to 0.                                       */
/*              -- If set, must be an integer number of seconds.         */
/*                 A positive value indicates that the value is AFTER    */
/*                 the Period of the ESDT.  A Negative value is          */
/*                 subtracted fromt he end of the period to find data    */
/*                 starting within the period specified for the ESDT.    */
/*         Example                                                       */
/*              END_PERIOD_OFFSET = "-7200" (2 hours)                    */
/***********************************************************************/

     END_PERIOD_OFFSET = 0

/***********************************************************************/
/*         Input file group ID                                          */
/*              -- Required for all PCF ENTRY objects with               */
/*                 PCF_FILE_TYPE =1, 3, 6 (ignored otherwise).           */
/*              -- Only used when input is defined as Static in ESDT     */
/*                 ODL.                                                  */
/*              -- Must be a string                                      */
/*              -- 1st character must be one of {C,L,D,O}                */
/*                    C -- Coefficient file                              */
/*                    L -- Lookup file                                   */
/*                    D -- Database file                                 */
/*                    O -- Other Type file                               */
/*              -- Rest of string must resolve to a                      */
/*                 positive integer < 10000                              */
```

```
/*          Example                                                 */
/*             SCIENCE_GROUP = "C1"                                 */
/*********************************************************************/

    SCIENCE_GROUP = ""

/*********************************************************************/
/*          Type of Input                                           */
/*             -- Required for all PCF ENTRY objects with        */
/*                PCF_FILE_TYPE = 1,3,6 (ignored otherwise)      */
/*             -- Must be a string with one of the following values:  */
/*                "Required" = Required input/no alternates        */
/*                "Primary" = Primary input/alternates defined      */
/*                            Alternate_Input object defined for this  */
/*                            PCF Entry.                           */
/*                "Optional" = Optional input, PGE can run without it.  */
/*                            An Optional_Input object must be defined */
/*                            for this PCF Entry.                    */
/*                "Alternate" = Alternate input/there will be an      */
/*                             Alternate_Input object defined for this */
/*                             PCF Entry.                           */
/*          Example                                                 */
/*             INPUT_TYPE = "Required"                              */
/*********************************************************************/

    INPUT_TYPE = ""

/*********************************************************************/
/*          Align DPR Time with Input                               */
/*             -- Specifies that the time of the DPR will be shifted  */
/*                to match the real time of input for this Logical Id.  */
/*             -- May only be set for one input per PGE Profile.    */
/*             -- Valid values are "Y" or "N".                      */
/*             -- If not specified, it is set to "N".               */
/*          Example                                                 */
/*             ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y"                 */
/*********************************************************************/

    ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"

/*********************************************************************/
/*          Number of Alternate Inputs needed.                      */
/*             -- Required for all PCF ENTRY objects with        */
/*                PCF_FILE_TYPE = 1,3,6 that have                */
/*             INPUT_TYPE = "Primary" (ignored otherwise)        */
/*             -- Must be either 0 or 1.                            */
/*          Example                                                 */
/*             NUMBER_NEEDED = 1                                    */
/*          (This means that only 1 of the alternate inputs is    */
/*             required to execute the PGE)                         */
/*********************************************************************/

    NUMBER_NEEDED =

/*********************************************************************/
/*          Distinct Value for the input.                           */
/*             -- Optional entry for PCF ENTRY objects with        */
/*                PCF_FILE_TYPE = 1,3,6.  Set to null if not provided.  */
/*             -- A string value, max length 80 characters.         */
```

```
/*             -- A value that will allow unqiue naming of granules   */
/*                input by a PGE.                          */
/*             -- Must be the name of a metadata parameter defined in   */
/*                a METADATA_DEFINITION objected.  If a parameter is   */
/*                is specified for which no METADATA_DEFINITION object   */
/*                exists an error will be raised during ODL parsing.  */
/*             -- Supports what are called Multi-Granule ESDTs.  These   */
/*                are ESDTs that have multiple granules for the same   */
/*                time period where the only difference between the   */
/*                granules is metadata parameters.            */
/*         Example                                    */
/*             DISTINCT_VALUE = "CAMERA_DF"                     */
/**********************************************************************/

    DISTINCT_VALUE = ""

/**********************************************************************/
/*         Query Type for the input.                              */
/*             -- Optional entry for PCF ENTRY objects with      */
/*                PCF_FILE_TYPE = 1,3,6.                     */
/*             -- Must be one of                          */
/*         "Temporal" -- Data is retrieved by time.          */
/*         "Spatial"  -- Data is retrieved by spatial location */
/*                   of 'key' data type.                  */
/*         "Tile"     -- Data is retrieved by spatial location */
/*                   of the tile.                       */
/*         "Already Created Tile"                       */
/*                 -- Data is retrieved by query of tiles      */
/*                    already produced (used for cases when */
/*                  one PGE needs the tile output of one or   */
/*                  more other PGEs).              */
/*         "Metadata" -- Data is retrieved via temporal query and*/
//                         a metadata query                          */
/*          -- NOTE that if "Already Created Tile" is used, then      */
/*             a Metadata Query is expected to query on the TileId   */
/*          parameter in the metadata.  "Already Created Tile"  */
/*          will NOT work without a metadata parameter that holds      */
/*          the TileId.                               */
/*          -- The default is "Temporal" (if not specified).        */
/*         Example                                    */
/*             QUERY_TYPE = "Temporal"                      */
/**********************************************************************/

    QUERY_TYPE = ""

/**********************************************************************/
/*         Spatial Time Delta.                               */
/*             -- Required for PCF ENTRY objects with         */
/*                PCF_FILE_TYPE = 1,3,6 that have QUERY_TYPE =    */
/*          "Spatial".                               */
/*              -- An Integer that allows for some time differential    */
/*          when querying for input data on spatial constraints.     */
/*          It is added to the Start/Stop times of the DPR.    */
/*          -- Time is specified in seconds               */
/*         Example                                    */
/*             SPATIAL_TIME_DELTA = 100                     */
/**********************************************************************/

    SPATIAL_TIME_DELTA =
```

```
/**********************************************************************/
/*          Spatial Pad                                             */
/*              -- Required for PCF ENTRY objects with              */
/*                 PCF_FILE_TYPE = 1,3,6 that have QUERY_TYPE =      */
/*                 "Temporal".                                      */
/*              -- A real number (float) value equal to 0.0 or 1000.0. */
/*                 Or, a value between those endpoints.  The units of */
/*                 measure is  kilometers.  INTEGERS are not valid!  */
/*                 (i.e. 10, 500)                                   */
/*              -- This pad will be applied to the KEY INPUT granule */
/*          Example                                                 */
/*              SPATIAL_PAD = 100.0                                 */
/**********************************************************************/

    SPATIAL_PAD =

/**********************************************************************/
/*          Key Input Data Type.                                    */
/*            -- Optional for PCF ENTRY objects with                */
/*                PCF_FILE_TYPE = 1,3,6 that have QUERY_TYPE =       */
/*            "Temporal" (ignored otherwise).                       */
/*            -- Specifies one of the following:                    */
/*                --     Spatial constaints of this input should be */
/*                   used when acquiring all data with QUERY_TYPE = */
/*                "Spatial".                                        */
/*                -- The number of granules for the input should    */
/*                 determine if a DataScheduled PGE should be       */
/*                 run.                                             */
/*            -- Must be one of "Y" or "N".                         */
/*            -- "YES" should only be set for a single input with a */
/*             QUERY_TYPE = "Temporal".                             */
/*            -- NOTE that the old version of this parameter        */
/*             SPATIAL_KEY_INPUT is still supported and will be     */
/*             treated as having the same meaning.                  */
/*          Example                                                 */
/*              KEY_INPUT = "Y"                                     */
/**********************************************************************/

    KEY_INPUT = ""

/**********************************************************************/
/* OPTIONAL PARAMETER                                               */
/*          Query Offset for Closest Granule.                       */
/*              -- Optional entry for PCF ENTRY objects with        */
/*                 PCF_FILE_TYPE = 1,3,6.  Set to 0 if not provided. */
/*              -- Must contain a single P=V string, where          */
/*                 P is one of {WEEKS, DAYS, HOURS, MINS, SECS}.     */
/*                 Other valid period values are NOT supported for this */
/*                 parameter.                                       */
/*              -- Used if input is expected to be the "Closest Granule".*/
/*                 This means that the data under this PCF_ENTRY will be */
/*                 queried for every CLOSEST_QUERY_OFFSET from the   */
/*                 Start Time of the Data Processing Request for the PGE,*/
/*                 either forward or backward as indicated by the value */
/*                 of CLOSEST_QUERY_DIRECTION.                      */
/*              -- Closest Granule supercedes Most Recent Granule   */
/*          Example                                                 */
/*              CLOSEST_QUERY_OFFSET = "DAYS=1"                     */
```

```
/**********************************************************************/

    CLOSEST_QUERY_OFFSET =

/**********************************************************************/
/*          Closest Granule Direction.                           */
/*              -- Required for PCF ENTRY objects with           */
/*                 PCF_FILE_TYPE = 1,3,6 that have specified      */
/*                 CLOSEST_QUERY_OFFSET.                          */
/*              -- A string that indicates the direction of a search */
/*                 for a desired granule.  Must be either:       */
/*                 "Forward"    or    "Backward"                 */
/*              -- CLOSEST_QUERY_DIRECTION determines the direction */
/*                 of search (timewise) to query for a suitable granule */
/*                 from the Start Time of the Data Processing Request */
/*                 for the PGE, either forward or backward.       */
/*              -- Closest Granule supercedes Most Recent Granule */
/*          Examples                                             */
/*              CLOSEST_QUERY_DIRECTION = "Forward"              */
/*              CLOSEST_QUERY_DIRECTION = "Backward"             */
/**********************************************************************/

    CLOSEST_QUERY_DIRECTION =

/**********************************************************************/
/*          Closest Granule Maximum Number of Retries.           */
/*              -- Required for PCF ENTRY objects with           */
/*                 PCF_FILE_TYPE = 1,3,6 that have specified      */
/*                 CLOSEST_QUERY_OFFSET.                          */
/*              -- An Integer that allows a number of retries on the */
/*                 inputs where the "Closest Granule" is expected. */
/*              -- The Query Offset set in the above parameter    */
/*                 (CLOSEST_QUERY_OFFSET) is used to repeat the   */
/*                 the query for the data for for time periods of */
/*                 Query Offset starting from the Start Time of the */
/*                 Data Processing Request for the PGE either forward or */
/*                 backward as indicated by the value            */
/*                 of CLOSEST_QUERY_DIRECTION.                    */
/*              -- Closest Granule supercedes Most Recent Granule */
/*          Example                                              */
/*              CLOSEST_QUERY_RETRIES = 20                       */
/**********************************************************************/

    CLOSEST_QUERY_RETRIES =

/**********************************************************************/
/* File Types Object                                        */
/*                                                  */
/* THIS OBJECT IS REQUIRED for PCF_FILE_TYPES = 1, 2, 3, 4, 5, 6. */
/*                                                  */
/* The default value for FILETYPE_NAME = "Single File Granule" is      */
/* usually all that is needed.   This means that the input/output only */
/* has one file per granule.  Note that this is separate from the */
/* MIN/MAX_GRANULES_REQUIRED and MIN/MAX_GRANULE_YIELD parameters which */
/* tell how many granules are desired for the PCF entry.          */
/*                                                  */
/* If the Data Type defined under this PCF entry can have multiple     */
/* files per data granule then this entry must be updated and there has */
/* to be a corresponding entry in the ESDT ODL file for this Data Type. */
```

```
/* There needs to be one of these File Type objects for every File Type */
/* associated with this PCF entry.  This object defines what file       */
/* type(s) this PGE wants to use for this PCF entry.                     */
/*                                                          */
/* Note that for L0 inputs, there should only be 1 File Type (different */
/* than "Single File Granule") that defines the number of files in a    */
/* L0 granule.                                              */
/*                                                          */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/***********************************************************************/

OBJECT = FILETYPE

/***********************************************************************/
/*          Class (object counter, used only to distinguish objects)    */
/*              -- Must be an integer                            */
/*              -- Must be unique in this file                   */
/*          Example                                          */
/*              CLASS = 1                                     */
/***********************************************************************/

   CLASS = 1

/***********************************************************************/
/*          Name of File Type.                                 */
/*              -- Must be a string, max len 40 characters.  Should     */
/*          be meaningful in that the name indicates what sort of     */
/*          data is stored within this file type.               */
/*              -- Defines what File Type is associated with this PCF   */
/*          entry.  It will determine how many entries are     */
/*          created under this logical ID in the PCF.           */
/*          Example                                          */
/*              FILETYPE_NAME  = "Instrument Band 7"            */
/***********************************************************************/

   FILETYPE_NAME = "Single File Granule"

/***********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/***********************************************************************/

END_OBJECT = FILETYPE

/***********************************************************************/
/* AUXILIARY LOGICAL ID object                                 */
/*                                                          */
/* Defines auxiliary logical Ids for a particular input.        */
/* This is used when there may be multiple granules for a particular    */
/* Logical Id and the PGE wants each granule under a separate logical   */
/* Id.  The best example of this is the case where a specific L0  */
/* input could have multiple granules satisfying the given time period. */
/* Since only 1 L0 granule is allowed per logical Id, Auxiliary Logical */
/* Ids can be used to spread the subsequent L0 granules among many      */
/* Logical IDs.                                               */
/*                                                          */
/* When Auxiliary Logical Ids are specified, the first granule that     */
/* satisfies the input requirements (time period, metadata checks,      */
/* etc.) will be placed under the Logical Id defined under the        */
/* PCF_ENTRY.  Each subsequent granule will be placed under an          */
```

```
/* Auxiliary Logical Id.  The granules are sorted by time, so the      */
/* earliest will go under the PCF_ENTRY Logical Id, with the           */
/* Auxiliary Logical Ids filled with later and later granules.         */
/*                                                  */
/* There can be more than one AUXILIARY_LOGICAL_ID per PCF_ENTRY,      */
/* and if there is one AUXILIARY_LOGICAL_ID object, then there has to  */
/* the same number as specified for MAX_GRANULES_REQUIRED.         */
/*                                                  */
/* This object is optional for PCF ENTRY objects with            */
/* PCF_FILE_TYPE = 1, 3 or 6(ignored otherwise).  If not needed, this  */
/* object should be deleted.                              */
/*                                                  */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/****************************************************************************/

   OBJECT = AUXILIARY_LOGICAL_ID

/****************************************************************************/
/*          Class (object counter, used only to distinguish objects)    */
/*             -- Must be an integer                         */
/*             -- Must be unique in this file for this type of object   */
/*             -- Must be greater than 0.                      */
/*          Example                                  */
/*             CLASS = 1                                */
/****************************************************************************/

      CLASS = 1

/****************************************************************************/
/*          Auxiliary Logical Id                         */
/*             -- The Logical Id to place subsequent granules under     */
/*          when creating the PCF.                        */
/*           -- Must be a positive integer.                   */
/*             -- The Ids specified for Toolkit use (10000 to 10999)    */
/*          will not be allowed.                         */
/*          Example:                                 */
/*             AUX_LOGICAL_ID = 1001                        */
/****************************************************************************/

      AUX_LOGICAL_ID =

/****************************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/****************************************************************************/

   END_OBJECT = AUXILIARY_LOGICAL_ID


/****************************************************************************/
/* Alternate Input object                              */
/*                                                  */
/* Defines parameter names and values for this Data Input to be   */
/* designated as an "alternate input."  This is defined as an input    */
/* that can be substituted for another, already defined input.         */
/*                                                  */
/* Note that the "Primary" or first choice Alternate input is also     */
/* designated an Alternate input and thus should have one of these     */
/* objects.  Order should be set to 1.  All subsequent Alternates      */
/* should have the same Alternate_Category as the primary and should   */
```

```
/* have Order > 1.                                               */
/*                                                     */
/* This object is optional for PCF ENTRY objects with           */
/* PCF_FILE_TYPE = 1, 3 or 6(ignored otherwise).  If not needed, this   */
/* object should be deleted.                           */
/*                                                     */
/* There can only be one of these objects per PCF ENTRY.        */
/*                                                     */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present      */
/***********************************************************************/

   OBJECT = ALTERNATE_INPUT

/***********************************************************************/
/*           Class (object counter, used only to distinguish objects)   */
/*              -- Must be an integer                           */
/*              -- Must be unique in this file for this type of object   */
/*              -- Must be greater than 0.                      */
/*           Example                                    */
/*              CLASS = 1                               */
/***********************************************************************/

      CLASS = 1

/***********************************************************************/
/*           Name of Alternate Category                         */
/*              -- Must be a string, max len 20 characters       */
/*              -- This is the grouping of Alternates for which this    */
/*           entry belongs.  The ORDER paramater defines which   */
/*               of the alternates is primary, secondary ...      */
/*             -- There should be at least one other entry with the */
/*               category.                              */
/*           Example:                                   */
/*             CATEGORY = "SeaSurfTemp"                   */
/***********************************************************************/

      CATEGORY = ""

/***********************************************************************/
/*         Default Order for this Alternate                      */
/*            Indicates the order of preference for alternates    */
/*          within the same category.                    */
/*            The primary (or first choice alternate) should have    */
/*          ORDER = 1.                                  */
/*             -- Must be an integer value.                 */
/*             -- Should be no greater than the maximum number of    */
/*               alternates for the specified CATEGORY.       */
/*           Example                                    */
/*             ORDER = 1 (this would be the primary alternate)     */
/***********************************************************************/

      ORDER =

/***********************************************************************/
/*         Runtime Parameter Logical Id for this Alternate.        */
/*            Sets up a runtime parameter (defined in the User   */
/*          Defined Runtime Parameters section of the PCF) that will */
/*            hold the logical ID of the chosen Alternate.       */
/*             -- Must be a positive integer value.          */
```

```
/*               -- Must NOT be a Toolkit specific logical ID        */
/*                  (10000 and 10999)                                */
/*               -- Must have a corresponding Runtime Parameter defined  */
/*                  in PCF section 5.                                */
/*           Example                                                 */
/*               RUNTIME_PARM_ID = 11111                             */
/*********************************************************************/

        RUNTIME_PARM_ID =

/*********************************************************************/
/*        Default Timer value to wait for Alternate to be avaiable   */
/*                -- Must contain a single P=V string, where         */
/*                   P is one of { MONTHS, WEEKS, DAYS, HOURS, MINS, SECS} */
/*                -- NOTE that this is not needed if WAITFOR (next    */
/*                   parameter) is set to "Y".                       */
/*           Example                                                 */
/*               TIMER = "DAYS=1"                                    */
/*********************************************************************/

        TIMER = "PV_Time_Value_goes_here"

/*********************************************************************/
/*        Wait For flag                                              */
/*             Informs PDPS to wait for the alternate input (regardless */
/*             of the timer value).  This means that even if the timer  */
/*           expires, PDPS will wait for it before executing the     */
/*           the PGE.                                                */
/*              -- A character value of either "Y" (YES) or "N" (NO).   */
/*             -- Must be set the same for all Alternates in the     */
/*              specified CATEGORY.  If one Alternate in the CATEGORY   */
/*              is set to "Y" then all WAITFOR flags for Alternates   */
/*              in that list also must have WAITFOR set to "Y".      */
/*           Example                                                 */
/*               WAITFOR = "N"                                       */
/*********************************************************************/

        WAITFOR = ""

/*********************************************************************/
/*        Temporal Flag                                              */
/*             Indicates if the alternate should be the previous     */
/*             incarnation of the Data Product (Y) rather than the   */
/*             most current Product (N).                             */
/*              -- A character value of either "Y" (YES) or "N" (NO).   */
/*           Example                                                 */
/*               TEMPORAL = "N"                                      */
/*********************************************************************/

        TEMPORAL = ""

/*********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present          */
/*********************************************************************/

   END_OBJECT = ALTERNATE_INPUT

/*********************************************************************/
/* Optional Input object                                             */
```

```
/*                                                   */
/* Defines parameter names and values for this Data Input to be   */
/* designated as an "optional input."  This means that it is an input   */
/* that is desired (if available), but that the PGE can process data    */
/* successfully without it.                              */
/*                                                   */
/* Note that Optional Inputs can work like Alternates, in that there   */
/* can be a selection to choose from and an order of preference.       */
/* In this case the first choice Optional input would be the "Primary"  */
/* (ORDER = 1).  If multiple Optional inputs are desired, it is best if */
/* they can be grouped as a list of "Primary" and its "Alternates".     */
/*                                                   */
/* This object is optional for PCF ENTRY objects with           */
/* PCF_FILE_TYPE = 1, 3 or 6 (ignored otherwise).              */
/*                                                   */
/* There can only be one of these objects per PCF ENTRY.        */
/* An input can either be Alternate or Optional, not both.          */
/* If a PCF entry is not Optional, this object should be deleted. */
/*                                                   */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present        */
/*************************************************************************/

   OBJECT = OPTIONAL_INPUT

/*************************************************************************/
/*        Class (object counter, used only to distinguish objects)   */
/*            -- Must be an integer                         */
/*            -- Must be unique in this file for this type of object   */
/*            -- Must be greater than 0.                       */
/*        Example                                    */
/*           CLASS = 1                                 */
/*************************************************************************/

      CLASS = 1


/*************************************************************************/
/*        Name of Optional Category                          */
/*            -- Must be a string, max len 40 characters       */
/*            -- This is the grouping of optional inputs (one or more) */
/*             for which this entry belongs.  The ORDER paramater      */
/*             defines which of the optionals is primary,       */
/*               secondary ... for the case where there is more than   */
/*           one optional input.                          */
/*        Example:                                   */
/*           CATEGORY = "SeaSurfTemp"                        */
/*************************************************************************/

      CATEGORY = ""

/*************************************************************************/
/*       Default Order for this Optional Input                    */
/*           Indicates the order of preference for optionals       */
/*         within the same category (when there is more than 1).     */
/*          The primary (or first choice optional) should have */
/*          ORDER = 1.                                 */
/*            -- Must be an integer value.                     */
/*            -- Should be no greater than the maximum number of      */
/*              optionals for the specified CATEGORY.             */
```

```
/*          Example                                              */
/*              ORDER = 1 (this would be the primary optional input or    */
/*                  for a single optional input)                   */
/***********************************************************************/

        ORDER =

/***********************************************************************/
/*         Runtime Parameter Logical Id for this Optional Input.       */
/*              Sets up a runtime parameter (defined in the User    */
/*            Defined Runtime Parameters section of the PCF) that will */
/*              hold the logical ID of the chosen Optional input.   */
/*              -- Must be a positive integer value.               */
/*              -- Must NOT be a Toolkit specific logical ID         */
/*                 (10000 and 10999)                              */
/*              -- Must have a corresponding Runtime Parameter defined */
/*                 in PCF section 5.                              */
/*          Example                                              */
/*              RUNTIME_PARM_ID = 11111                            */
/***********************************************************************/

        RUNTIME_PARM_ID =

/***********************************************************************/
/*         Default Timer value to wait for Alternate to be avaiable    */
/*              -- Must contain a single P=V string, where         */
/*                 P is one of { MONTHS, WEEKS, DAYS, HOURS, MINS, SECS} */
/*              -- NOTE that this is not needed if WAITFOR (next     */
/*                 parameter) is set to "Y".                        */
/*          Example                                              */
/*              TIMER = "DAYS=1"                                   */
/***********************************************************************/

        TIMER = "PV_Time_Value_goes_here"

/***********************************************************************/
/*         Temporal Flag                                          */
/*              Indicates if the alternate should be the previous    */
/*              incarnation of the Data Product (Y) rather than the  */
/*              most current Product (N)                           */
/*              -- A character value of either "Y" (YES) or "N" (NO).  */
/*          Example                                              */
/*              TEMPORAL = "N"                                     */
/***********************************************************************/

        TEMPORAL = ""

/***********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/***********************************************************************/

    END_OBJECT = OPTIONAL_INPUT

/***********************************************************************/
/* Metadata checks object                                       */
/*                                              */
/* Defines parameter names, values and conditions for which this PGE  */
/* should execute if true for this input file                   */
/* PGE depends.                                             */
```

```
/*                                                    */
/* This object is optional for PCF ENTRY objects with            */
/* PCF_FILE_TYPE = 1,3 or 6 (ignored otherwise).  Delete if not needed. */
/* Replicate object if multiple METADATA_CHECKS are required.         */
/*                                                    */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present        */
/***********************************************************************/

   OBJECT = METADATA_CHECKS


/***********************************************************************/
/*         Class (object counter, used only to distinguish objects)   */
/*            -- This line is generated by DpAtCreateOdlTemplate */
/*               from the PCF and is normally not modified       */
/*            -- Must be an integer                              */
/*            -- Must be unique in this file for this type of object   */
/*            -- Must be greater than 0.                          */
/*         Example                                     */
/*            CLASS = 1                                 */
/***********************************************************************/

      CLASS = 1


/***********************************************************************/
/*         Name of metadata parameter on which this PGE depends    */
/*             -- Must be a string, max len 40 characters.       */
/*             -- Must be present in the ESDT ODL file for this ESDT.   */
/*             -- Means that the specified metadata parameter must have */
/*                the specified value for the PGE to execute.       */
/*            -- For Product Specific Attibutes (PSAs), this is the     */
/*             name of the attribute in question.  The corresponding */
/*             entry the ESDT_ODL file must specify CONTAINER_NAME =     */
/*             "AdditionalAttributes".                      */
/*         Example:                                    */
/*          The PGE depends on the metadata value for the parameter */
/*          called "tbd_parm_name".                        */
/*            PARM_NAME = "tbd_parm_name"                      */
/***********************************************************************/

      PARM_NAME = ""


/***********************************************************************/
/*         Operator for dependency condition                  */
/*             -- Must be one of { >, <, >=, <=, ==, != }       */
/*           -- This means that the metadata parameter is:       */
/*               ">" -- actual parameter value must be greater than    */
/*                    value specified in VALUE.                 */
/*               "<" -- actual parameter value must be less than */
/*                    value specified in VALUE.                 */
/*               ">=" -- actual parameter value must be greater than    */
/*                    or equal to value specified in VALUE.        */
/*               "<=" -- actual parameter value must be less than or    */
/*                    equal to value specified in VALUE.        */
/*               "==" -- actual parameter value must be equal to */
/*                    value specified in VALUE.                 */
/*               "!=" -- actual parameter value must be NOT equal to    */
/*                    value specified in VALUE.                 */
/*         Example                                     */
/*            OPERATOR = "=="                               */
```

```
/*************************************************************************/

        OPERATOR = ""

/*************************************************************************/
/*          Value for metadata parameter upon which this PGE depends     */
/*             -- The value for the metadata parameter that is to be      */
/*             checked against.                                    */
/*              -- Computer data type (string, float or long) of the     */
/*                 value must correspond to the computer data type */
/*                 given in the ESDT ODL file                        */
/*          Example                                          */
/*              VALUE = 0                                    */
/*          Requires that TYPE = "INT" for the "tbd_parm_name" object    */
/*          in ODL file                                      */
/*           $DPAT_ESDT_SCIENCE_MD/ESDT_<ESDTName#Version>.odl     */
/*              VALUE = "Joe"                                  */
/*          Requires that TYPE = "STR" for the "tbd_parm_name" object    */
/*          in ODL file                                      */
/*           $DPAT_ESDT_SCIENCE_MD/ESDT_<ESDTName#Version>.odl     */
/*************************************************************************/

        VALUE = ""

/*************************************************************************/
/*          Database query Value                               */
/*             -- OPTIONAL parameter.  Defaults to "NONE".        */
/*            -- Set to define this Metadata Query as having a    */
/*             a VALUE set by PDPS based on the run of the PGE.   */
/*             This Metadata Query will then be performed on the */
/*             value retrieved from the PDPS database rather than */
/*             the value specified in the VALUE parameter.       */
/*              -- Must be one of {"NONE", "PATH NUMBER",        */
/*             "ORBIT NUMBER", "TILE ID", "START DATA DAY",       */
/*             "END DATA DAY", "ORBIT IN DAY", "GRANULE IN ORBIT", */
/*             "YEAR OF DATA", "MONTH OF DATA", "DAY OF DATA"}     */
/*             "NONE" -- no dynamic value, use VALUE             */
/*             "PATH NUMBER" -- get the orbital path number      */
/*             "ORBIT NUMBER" -- get the number of the orbit     */
/*             "TILE ID" -- get the id of the tile              */
/*             "START DATA DAY" -- get the start data day       */
/*                "END DATA DAY" -- get the end data day        */
/*                "ORBIT IN DAY" -- get the orbit number within day    */
/*             "GRANULE IN ORBIT" -- get the granule within the   */
/*                          orbit assuming 6 minute             */
/*             "YEAR OF DATA" -- the year of the data       */
/*             "MONTH OF DATA" -- the month of the data          */
/*             "DAY OF DATA" -- the day of the data             */
/*          Example                                          */
/*             DATABASE_QUERY = "PATH NUMBER"                    */
/*************************************************************************/

    DATABASE_QUERY = "NONE"

/*************************************************************************/
/*  Optional Parameter.  Defaults to empty string if not specified.    */
/*                                              */
/*          Name of metadata parameter which provides a key into a      */
/*       a multi-containered object.  Such an object is the        */
```

```
/*          MeasuredParameters group in the inventory metadata.          */
/*               -- Must be a string, max len 40 characters.        */
/*               -- Must be present in the ESDT ODL file for this ESDT.   */
/*               -- Is matched with KEY_PARAMETER_VALUE to determine      */
/*                  the entry in a mult-containered metadata group. */
/*             -- For Product Specific Attibutes (PSAs), this entry       */
/*              should NOT be specified.                          */
/*             -- Because of Metadata Query limitations, there can only   */
/*              be one KEY_PARAMETER_NAME/KEY_PARAMETER_VALUE pair  */
/*              per PGE ODL File.  This is because only a single     */
/*              Metadata Query is allowed against the               */
/*              MeasuredParameters group.                         */
/*             -- For Metadata Queries within the MeasuredParameters      */
/*              group this should be set to the metadata field called     */
/*              "ParameterName".                                 */
/*          Example:                                         */
/*             KEY_PARAMETER_NAME = "ParameterName"              */
/**********************************************************************/

        KEY_PARAMETER_NAME = ""


/**********************************************************************/
/*  Optional Parameter.  Must be preset if KEY_PARAMETER_NAME exists.   */
/*  Defaults to the empty string if not specified.              */
/*                                           */
/*          Value of metadata parameter which provides a key into a      */
/*       a multi-containered object.  Such an object is the              */
/*       MeasuredParameters group in the inventory metadata.          */
/*               -- Must be a string, max len 80 characters.        */
/*               -- Must be present in the ESDT ODL file for this ESDT.   */
/*               -- Is matched with KEY_PARAMETER_NAME to determine      */
/*                  the entry in a mult-containered metadata group. */
/*             -- For Product Specific Attibutes (PSAs), this entry       */
/*              should NOT be specified.                          */
/*             -- Because of Metadata Query limitations, there can only   */
/*              be one KEY_PARAMETER_NAME/KEY_PARAMETER_VALUE pair  */
/*              per PGE ODL File.  This is because only a single     */
/*              Metadata Query is allowed against the               */
/*              MeasuredParameters group.                         */
/*             -- For Metadata Queries within the MeasuredParameters      */
/*              group this should be set to the desired value of the     */
/*              metadata field called "ParameterName".          */
/*          Example:                                         */
/*             KEY_PARAMETER_VALUE = "LandCoverage"              */
/**********************************************************************/

        KEY_PARAMETER_VALUE = ""


/**********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/**********************************************************************/

    END_OBJECT = METADATA_CHECKS


/**********************************************************************/
/* Metadata Query Object                              */
/*                                           */
/* Defines parameter names, values and conditions for which this Input  */
/* for the PGE should be selected.  Only data that matches the        */
```

A-25   611-EMD-001

```
/* with the specified metadata parameter with the specified value and   */
/* condition will be chosen as input to this PGE.  Note that if no       */
/* matching data if found the PGE will NOT execute.                      */
/*                                                            */
/* This object is optional for PCF ENTRY objects with                    */
/* PCF_FILE_TYPE = 1,3 or 6 (ignored otherwise).  Delete if not needed. */
/* Replicate object if multiple METADATA_QUERYs are required.            */
/*                                                            */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present      */
/***********************************************************************/

   OBJECT = METADATA_QUERY

/***********************************************************************/
/*          Class (object counter, used only to distinguish objects)    */
/*             -- This line is generated by DpAtCreateOdlTemplate */
/*                from the PCF and is normally not modified         */
/*             -- Must be an integer                              */
/*             -- Must be unique in this file for this type of object   */
/*             -- Must be greater than 0.                             */
/*          Example                                            */
/*             CLASS = 1                                       */
/***********************************************************************/

      CLASS = 1

/***********************************************************************/
/*          Name of metadata parameter on which this PGE depends    */
/*             -- Must be a string, max len 40 characters        */
/*             -- Must be present in the ESDT ODL file for this ESDT    */
/*          Example:                                            */
/*           This CERES PGE depends on the Q/A value of             */
/*                this ESDT "TRWpcal182": execute the CERES PGE only  */
/*                if ESDT "TRWpcal182" had Q/A parameter         */
/*                "tbd_parm_name" = 0                            */
/*             PARM_NAME = "tbd_parm_name"                       */
/***********************************************************************/

      PARM_NAME = "Parm_name_goes_here"

/***********************************************************************/
/*          Operator for dependency condition                        */
/*             -- Must be one of { >, <, >=, <=, ==, != }        */
/*          Example                                            */
/*             OPERATOR = "=="                                 */
/***********************************************************************/

      OPERATOR = "Operator_goes_here"

/***********************************************************************/
/*          Value for ESDT parameter upon which this PGE depends        */
/*             -- Computer data type (string, float or long) of the    */
/*                value must correspond to the computer data type */
/*                given in the ESDT ODL file                    */
/*          Example                                            */
/*             VALUE = 0                                       */
/*          Requires that TYPE = "INT" for the "tbd_parm_name" object    */
/*          in ODL file                                         */
/*           $DPAT_ESDT_SCIENCE_MD/ESDT_<ESDTName#Version>.odl      */
```

```
/*                VALUE = "Joe"                                    */
/*         Requires that TYPE = "STR" for the "tbd_parm_name" object   */
/*         in ODL file                                             */
/*          $DPAT_ESDT_SCIENCE_MD/ESDT_<ESDTName#Version>.odl       */
/*******************************************************************/

      VALUE = "Value_goes_here"


/*******************************************************************/
/*         Database query Value                                 */
/*            -- OPTIONAL parameter.  Defaults to "NONE".        */
/*          -- Set to define this Metadata Query as having a     */
/*           a VALUE set by PDPS based on the run of the PGE.    */
/*           This Metadata Query will then be performed on the   */
/*           value retrieved from the PDPS database rather than  */
/*           the value specified in the VALUE parameter.         */
/*            -- Must be one of {"NONE", "PATH NUMBER",          */
/*           "ORBIT NUMBER", "TILE ID", "START DATA DAY",        */
/*           "END DATA DAY", "ORBIT IN DAY", "GRANULE IN ORBIT", */
/*           "YEAR OF DATA", "MONTH OF DATA", "DAY OF DATA"}      */
/*           "NONE" -- no dynamic value, use VALUE               */
/*           "PATH NUMBER" -- get the orbital path number        */
/*           "ORBIT NUMBER" -- get the number of the orbit       */
/*           "TILE ID" -- get the id of the tile                 */
/*           "START DATA DAY" -- get the start data day          */
/*             "END DATA DAY" -- get the end data day            */
/*             "ORBIT IN DAY" -- get the orbit number within day */
/*           "GRANULE IN ORBIT" -- get the granule within the    */
/*                        orbit assuming 6 minute               */
/*           "YEAR OF DATA" -- the year of the data              */
/*           "MONTH OF DATA" -- the month of the data            */
/*         Example                                               */
/*           DATABASE_QUERY = "PATH NUMBER"                      */
/*******************************************************************/

    DATABASE_QUERY = "NONE"


/*******************************************************************/
/*  Optional Parameter.  Defaults to empty string if not specified.  */
/*                                                               */
/*         Name of metadata parameter which provides a key into a    */
/*      a multi-containered object.  Such an object is the       */
/*      MeasuredParameters group in the inventory metadata.      */
/*          -- Must be a string, max len 40 characters.          */
/*          -- Must be present in the ESDT ODL file for this ESDT. */
/*          -- Is matched with KEY_PARAMETER_VALUE to determine  */
/*            the entry in a mult-containered metadata group.    */
/*         -- For Product Specific Attibutes (PSAs), this entry  */
/*          should NOT be specified.                             */
/*         -- For Metadata Checks within the MeasuredParameters  */
/*          group this should be set to the metadata field called */
/*          "ParameterName".                                     */
/*         Example:                                              */
/*          KEY_PARAMETER_NAME = "ParameterName"                 */
/*******************************************************************/

      KEY_PARAMETER_NAME = ""


/*******************************************************************/
```

```
/*  Optional Parameter.  Must be preset if KEY_PARAMETER_NAME exists.   */
/*  Defaults to the empty string if not specified.                 */
/*                                                       */
/*          Value of metadata parameter which provides a key into a       */
/*       a multi-containered object.  Such an object is the          */
/*       MeasuredParameters group in the inventory metadata.          */
/*              -- Must be a string, max len 80 characters.       */
/*              -- Must be present in the ESDT ODL file for this ESDT.   */
/*              -- Is matched with KEY_PARAMETER_NAME to determine     */
/*                 the entry in a mult-containered metadata group. */
/*           -- For Product Specific Attibutes (PSAs), this entry      */
/*            should NOT be specified.                       */
/*           -- For Metadata Checks within the MeasuredParameters     */
/*            group this should be set to the desired value of the     */
/*            metadata field called "ParameterName".          */
/*          Example:                              */
/*           KEY_PARAMETER_VALUE = "LandCoverage"               */
/***********************************************************************/

        KEY_PARAMETER_VALUE = ""


/***********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/***********************************************************************/


    END_OBJECT = METADATA_QUERY


/***********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/***********************************************************************/


END_OBJECT = PCF_ENTRY



/***********************************************************************/
/* After this point, the comments only address unique parameters that   */
/* have not been explained above                        */
/*                                           */
/* Note that the order of PCF entries in not really inportant.  These   */
/* have been ordered the same as their order would be in the PGEs PCF.  */
/***********************************************************************/



OBJECT = PCF_ENTRY
    CLASS = 2
    LOGICAL_ID = 3000
    PCF_FILE_TYPE = 2
    DATA_TYPE_NAME = ""


/***********************************************************************/
/*          Minimum number of output granules for this logical ID  */
/*              -- This line is generated by DpAtCreateOdlTemplate */
/*                 from the PCF and is only modified if the PGE may     */
/*             successfully produce less granules than specified in     */
/*             the PCF used to generate the template.        */
/*              -- Required for all PCF ENTRY objects with        */
/*                 PCF_FILE_TYPE = 2, 4, 7 (ignored otherwise).      */
/*              -- Must be a positive integer.                */
/*              -- Note that for number of files within a granule    */
```

```
/*             greater than one, the FILE TYPE object for this entry    */
/*             must be changed to specify the various file types and     */
/*             maximum number of files.                                  */
/*          Example                                                      */
/*             MIN_GRANULE_YIELD = 1                                     */
/**********************************************************************/

    MIN_GRANULE_YIELD = 1


/**********************************************************************/
/*          Maximum number of output granules for this logical ID  */
/*             -- This line is generated by DpAtCreateOdlTemplate */
/*                from the PCF and is only modified if the PGE may     */
/*             successfully produce more granules than specified in    */
/*             the PCF used to generate the template.        */
/*              -- Required for all PCF ENTRY objects with         */
/*                PCF_FILE_TYPE = 2, 4, 7 (ignored otherwise).      */
/*              -- Must be a positive integer.                     */
/*           -- Note that for number of files within a granule    */
/*             greater than one, the FILE TYPE object for this entry    */
/*             must be changed to specify the various file types and    */
/*             maximum number of files.                                 */
/*          Example                                                     */
/*             MAX_GRANULE_YIELD = 1                                    */
/**********************************************************************/

    MAX_GRANULE_YIELD = 1


/**********************************************************************/
/*          Associated MCF ID                                          */
/*             -- The Logical ID of the MCF associated with this input. */
/*                Informs Data Processing as to the logical id which    */
/*                the PGE associates the MCF for this output.      */
/*             -- Required for all PCF ENTRY objects with          */
/*                PCF_FILE_TYPE = 2, 4 (if not output by the Toolkit),  */
/*                7.  (ignored otherwise).                          */
/*             -- Must be a positive integer.                      */
/*             -- NOTE that any input PCF entries that were created     */
/*                by CreateOdlTemplate for MCFs should be deleted.  The */
/*                information about which Logical IDs are for MCFs is   */
/*                is captured by this parameter for each output that    */
/*                the MCF is associated with.                       */
/*          Example                                                     */
/*             ASSOCIATED_MCF_ID  = 3001                                */
/**********************************************************************/

    ASSOCIATED_MCF_ID =


/**********************************************************************/
/*          Output file group ID                                       */
/*             -- Required for all PCF ENTRY objects with          */
/*                PCF_FILE_TYPE = 2 (ignored otherwise)            */
/*             -- Must be a string                          */
/*             -- 1st character must be one of {S,Q,H,B}           */
/*                S -- Science file                                */
/*                Q -- Q/A file                            */
/*                H -- Production history file                     */
/*                B -- Browse file                                */
/*             -- Rest of string must resolve to a               */
```

```
/*              positive integer < 1000                          */
/*          Example                                              */
/*            SCIENCE_GROUP = "S1"                               */
/*          Files associated with this science file would have       */
/*          SCIENCE_GROUP = "Q1", SCIENCE_GROUP = "B1", etc.         */
/**********************************************************************/

    SCIENCE_GROUP = " "

/**********************************************************************/
/*          Nominal no. of file instances with *different* logical IDs,  */
/*          but which are associated with each other                 */
/*              -- Optional for all PCF ENTRY objects with           */
/*                 PCF_FILE_TYPE = 2 (ignored otherwise).            */
/*              -- If 0, ignore this parameter -- no other logical IDs   */
/*                 are associated with it.                           */
/*          Example                                                  */
/*            INSTANCE = 0                                           */
/*          Note: This parameter is specifically designed to accomodate  */
/*          the CERES case where 24 standard product files are generated */
/*          per day, each with a *different* logical ID, but are all     */
/*          essentially an instance of a single file format              */
/*          In this case INSTANCE would take values 1, 2, ..., 24   */
/**********************************************************************/

    INSTANCE = 0

/**********************************************************************/
/*          Distinct Value for the output.                          */
/*              -- Optional entry for PCF ENTRY objects with         */
/*                 PCF_FILE_TYPE = 2,4,7.  Set to null if not provided.  */
/*              -- A string value, max length 80 characters.         */
/*            -- A value that will allow unqiue naming of granules       */
/*             produced by a PGE.                                   */
/*            -- Must be the name of a metadata parameter defined in     */
/*             a METADATA_DEFINITION objected.  If a parameter is   */
/*             is specified for which no METADATA_DEFINITION object      */
/*             exists an error will be raised during ODL parsing.   */
/*            -- Supports what are called Multi-Granule ESDTs.  These    */
/*                are ESDTs that have multiple granules for the same    */
/*             time period where the only difference between the    */
/*             granules is metadata parameters.                     */
/*          Example                                                 */
/*            DISTINCT_VALUE = "CAMERA_DF"                          */
/**********************************************************************/

    DISTINCT_VALUE = ""

/**********************************************************************/
/*          Minimum expected size (in MB) of this output            */
/*            (used for QA purposes).                               */
/*              -- Required for all PCF ENTRY objects with           */
/*                 PCF_FILE_TYPE = 2 (ignored otherwise)            */
/*              -- Must be a positive integer                       */
/*          Example                                                 */
/*            MINIMUM_SIZE = 120000                                 */
/**********************************************************************/

    MINIMUM_SIZE = 0
```

```
/**********************************************************************/
/*          Maximum expected size (in MB) of this output             */
/*            (used for QA purposes).                                */
/*              -- Required for all PCF ENTRY objects with           */
/*                 PCF_FILE_TYPE = 2 (ignored otherwise)             */
/*              -- Must be a positive integer                        */
/*              -- Must be larger than or equal to MINIMUM_SIZE      */
/*          Example                                                   */
/*            MAXIMUM_SIZE = 50000000                                 */
/**********************************************************************/

   MAXIMUM_SIZE = 1

   OBJECT = FILETYPE
      CLASS = 1
      FILETYPE_NAME = "Single File Granule"
   END_OBJECT = FILETYPE


/**********************************************************************/
/* Associated Science Data Object                         */
/*                                             */
/* THIS OBJECT IS REQUIRED for Outputs where the SCIENCE_GROUP       */
/* contains 'B' or 'Q' (meaning it is a BROWSE or QA granule).  It is */
/* ignored otherwise.                                      */
/*                                             */
/* BROWSE and QA output granules are linked to the science granules  */
/* for which they are produced.  This linkage occrus when the produced */
/* BROWSE or QA granules are inserted to the Data Server.  This object */
/* defines the linkage so that the correct link can be made after */
/* the PGE completes and its outputs are inserted to the Data Server. */
/*                                             */
/* If more than one science granule is associated with the BROWSE or */
/* QA output defined by this PCF_ENTRY, then repeat the Associated    */
/* Science Data Objects to specify the various Logical Ids that define */
/* those Associated Science Granules.                       */
/*                                             */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/**********************************************************************/

OBJECT = ASSOCIATED_SCIENCE_DATA

/**********************************************************************/
/*          Class (object counter, used only to distinguish objects) */
/*              -- Must be an integer                         */
/*              -- Must be unique in this file                */
/*          Example                                          */
/*            CLASS = 1                                      */
/**********************************************************************/

   CLASS = 1

/**********************************************************************/
/*          Associated Science Granule's Logical Id            */
/*              -- Must a positive integer value.              */
/*              -- Defines which logical Id this BROWSE/QA granules is */
/*          Associated with.  This means that when the associated     */
/*          science granule is inserted to the Data Server, a    */
/*          will be made with the BROWSE/QA granule defined by  */
```

```
/*              this PCF_ENTRY.                        */
/*          -- A check will be done to verify that the Logical ID    */
/*              has been defined in the ODL file.              */
/*          Example                                  */
/*             ASSOCIATED_SCIENCE_LOGICAL_ID = 12345           */
/***********************************************************************/

   ASSOCIATED_SCIENCE_LOGICAL_ID =

/***********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present     */
/***********************************************************************/

END_OBJECT = ASSOCIATED_SCIENCE_DATA

END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY

   CLASS = 3
   LOGICAL_ID = 200

/***********************************************************************/
/* This is an example for in Support input.                        */
/***********************************************************************/
   PCF_FILE_TYPE = 3

/***********************************************************************/
/* Support input and output types (if not associated with generic    */
/* Toolkit files) have their own Data Types and Versions.         */
/***********************************************************************/

   DATA_TYPE_NAME = ""
   DATA_TYPE_VERSION = ""

/***********************************************************************/
/* This is always 1 for non-product inputs                        */
/***********************************************************************/
   DATA_TYPE_REQUIREMENT = 1

/***********************************************************************/
/* Support inputs can be any input type.  Though none are       */
/* shown, they can have Alternate or Optional input objects as well    */
/* Metadata checks objects.                           */
/***********************************************************************/
   INPUT_TYPE = ""
   NUMBER_NEEDED = 1

   OBJECT = FILETYPE
      CLASS = 1
      FILETYPE_NAME = "Single File Granule"
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY

   CLASS = 4
   LOGICAL_ID = 4000
   PCF_FILE_TYPE = 5
```

```
/**********************************************************************/
/*          Runtime parameter name                            */
/*              -- This line is generated by DpAtCreateOdlTemplate */
/*                 from the PCF and is normally not modified       */
/*              -- Required for all PCF ENTRY objects with         */
/*                 PCF_FILE_TYPE = 5 (ignored otherwise)               */
/*              -- Must be a string, max len 50 characters         */
/*          Example                                        */
/*              PGE_PARAMETER_NAME = "Spacecraft_Class"            */
/**********************************************************************/

    PGE_PARAMETER_NAME = " "

/**********************************************************************/
/*          Runtime parameter default value                    */
/*              -- This line is generated by DpAtCreateOdlTemplate */
/*                 from the PCF and is normally not modified       */
/*              -- Required for all PCF ENTRY objects with         */
/*                 PCF_FILE_TYPE = 5 (ignored otherwise)               */
/*              -- Must be a string, max len 200 characters        */
/*            -- If double quotes must be included in the string   */
/*            (i.e. the string must read "This is the string")     */
/*            then single quotes must be placed around the string.     */
/*            Thus "This is the string" would become '"This is the     */
/*            string"'.  Note that this automatically done by      */
/*            the CreateODLTemplate tool.                       */
/*          Example                                        */
/*              PGE_PARAMETER_DEFAULT = "TRMM"                      */
/**********************************************************************/

    PGE_PARAMETER_DEFAULT = " "

/**********************************************************************/
/*          Runtime parameter Dynamic Value                    */
/*              -- This line is generated by DpAtCreateOdlTemplate */
/*                 from the PCF and is set to "NONE".                  */
/*            -- Set to define this runtime parameter as having a  */
/*            a value set by PDPS based on the run of the PGE.   */
/*            This runtime parameter will then have the value of  */
/*            the specified attribute when the PCF is created.   */
/*             -- Required for all PCF ENTRY objects with         */
/*                PCF_FILE_TYPE = 5 (ignored otherwise)               */
/*             -- Must be one of {"NONE", "PATH NUMBER",            */
/*            "ORBIT NUMBER", "TILE ID", "START DATA DAY",          */
/*            "END DATA DAY"}                           */
/*            "NONE" -- no dynamic value, use Default        */
/*            "PATH NUMBER" -- get the orbital path number          */
/*            "ORBIT NUMBER" -- get the number of the orbit         */
/*            "TILE ID" -- get the id of the tile                   */
/*            "START DATA DAY" -- get the start data day            */
/*             "END DATA DAY" -- get the end data day               */
/*          Example                                        */
/*              PGE_PARAMETER_DYNAMIC_VALUE = "PATH NUMBER"         */
/**********************************************************************/

    PGE_PARAMETER_DYNAMIC_VALUE = "NONE"


/**********************************************************************/
```

```
/*         Profile Selector Runtime Parameter Flag            */
/*           -- This line is generated by DpAtCreateOdlTemplate */
/*             from the PCF and is set to "N".                 */
/*           -- Must be a string, value of either "Y" (for Yes) and    */
/*             "N" (for No).                                   */
/*           -- If not specified, defaults to "N".             */
/*           -- Indicates that this Runtime Parameter (along with      */
/*            others) uniquely identifies a profile of this PGE   */
/*              (PGE Name + PGE version) based on the PARAMETER_NAME  */
/*            and DEFAULT_VALUE pair.                          */
/*            -- If set to "Y" for any Runtime Parameter, then the     */
/*               RegisterPGE tool will check to make sure that this    */
/*               Runtime Parameter/Default Value pairs flaged          */
/*             assures that this PGE Profile is different from all */
/*             the rest.                                       */
/*          Example                                            */
/*            PROFILE_SELECTOR_PGE_PARAMETER = "N"             */
/***********************************************************************/

   PROFILE_SELECTOR_PGE_PARAMETER = ""

END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY

   CLASS = 5
   LOGICAL_ID = 200


/***********************************************************************/
/* This is an example for an Interim/Intermediate input.            */
/***********************************************************************/
   PCF_FILE_TYPE = 6


/***********************************************************************/
/* Interim/Intermediate input and output types have their own Data    */
/* Types and Versions.                                             */
/***********************************************************************/

   DATA_TYPE_NAME = ""
   DATA_TYPE_VERSION = ""


/***********************************************************************/
/*         Last PGE to Use Interim Data Type?                   */
/*         This is a "Y" or "N" parameter that defines if this PGE     */
/*         (the one defined by this ODL file) is the last to use this  */
/*         Interim Data type.                                   */
/*            -- Must be a string or "Y" or "N".               */
/*         Example                                             */
/*            INTERIM_LAST_PGE_TO_USE = "N"                     */
/***********************************************************************/

   INTERIM_LAST_PGE_TO_USE  = "N"

   DATA_TYPE_REQUIREMENT = 1


/***********************************************************************/
/* Interim/Intermediate inputs can be any input type.  Though none are */
/* are shown, they can have Alternate or Optional input objects as well */
/* Metadata checks objects.                                        */
```

```
/*******************************************************************/
   INPUT_TYPE = ""
   NUMBER_NEEDED = 1

   OBJECT = FILETYPE
      CLASS = 1
      FILETYPE_NAME = "Single File Granule"
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

/*******************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED                      */
/*******************************************************************/


END
```

A.1.2 ESDT_ODL.template

```
/*******************************************************************/
/*                                                  */
/*          TEMPLATE ESDT SCIENCE METADATA ODL FILE         */
/*                                                  */
/*                                                  */
/* The SSIT operator's responsibility is to copy this file over and    */
/* edit it to add all necessary PDPS metadata values.          */
/*                                                  */
/* Each ESDT used by a PGE must have a corresponding ESDT SCIENCE */
/* metadata ODL file.                               */
/*                                                  */
/* All ESDT ODL files must reside in directory $DPAT_ESDT_SCIENCE_MD .  */
/*                                                  */
/* The operator must add a value to the right of the "=" for each       */
/* parameter.                                       */
/*                                                  */
/* CHANGE LOG                                       */
/*    -- Added File Type object.                     05/28/97  */
/*    -- Added Processing Level.                     06/04/97  */
/*    -- Added Orbit types to period/boundary comments.    06/07/97  */
/*    -- Updated Archived_By and Processed_By to be             */
/*       required for all types but Static.          06/24/97  */
/*    -- Allowed for 0 value in Interim Short Delete.   10/07/97  */
/*    -- Added DURATION parameter.                   11/14/97  */
/*    -- Removed OVERLAP as a choice for prediction meth.  11/03/97  */
/*    -- Changed METADATA_CHECKS to METADATA_DEFINITION.   12/06/97  */
/*       Updated description of FILETYPE object.               */
/*    -- Added optional METADATA_CONTAINER parameter.   12/15/97  */
/*    -- Added info on Metadata_Defintion for        02/04/98    */
/*          Product Specific Attibutes.                        */
/*    -- Added The Distinct Parameter definition.        03/24/98    */
/*    -- Fixed length for PROVIDER, FILETYPE_NAME.       03/31/98    */
/*    -- Made CONTAINER no longer optional for METADATA   05/06/98    */
/*          DEFINTION object.                         */
/*    -- Updated definition of USE_OBJECT.               06/25/98    */
/*    -- Added KEY_PARAMATER_NAME and KEY_PARAMETER_VALUE  07/05/98    */
/*          for Metadata Definition objects.                   */
/*          Updated description for DISTINCT_PARAMETER.        */
/*    -- Updated lengths for INSTRUMENT and PLATFORM.     08/13/98    */
```

```
/*    -- Added "NONROUTINE" for PREDICTION_METHOD          08/24/98    */
/*          parameter.  This is for ASTER Routine Processing.   */
/*    -- Added PRODUCTION_CHAIN object                      07/12/99    */
/*      -- Added ONDEMAND_DELETION_INTERVAL parameter              */
/*                                                            */
/***********************************************************************/


/***********************************************************************/
/*          Data Type name                                  */
/*              -- Must be a string, max len 8 characters        */
/*              -- ESDT name inside ODL file must be identical to  */
/*                      ESDT name used as part of ODL filename,        */
/*                      which in turn was generated from the        */
/*                      DATA_TYPE_NAME in the PGE ODL file for the PCF    */
/*                      entry.                                */
/*              -- It should be the same as the Short Name used in the  */
/*                      ESDT defintion at the Data Server.           */
/*          Example                                       */
/*              DATA_TYPE_NAME  = "NMC"                       */
/***********************************************************************/

DATA_TYPE_NAME = ""


/***********************************************************************/
/*          Data Type Version                               */
/*              -- Must be a string, max len 5 characters        */
/*              -- ESDT name inside ODL file must be identical to  */
/*                      ESDT name used as part of ODL filename,        */
/*                      which in turn was generated from the        */
/*                      DATA_TYPE_VERSION in the PGE ODL file for the PCF  */
/*                      entry.                                */
/*              -- It should be the same as the VersionID used in the   */
/*                      ESDT defintion at the Data Server.            */
/*              -- Note that this is not important for Interim/      */
/*                  Intermediate types.                      */
/*          Example                                       */
/*              DATA_TYPE_VERSION  = "3.5.1"                    */
/***********************************************************************/

DATA_TYPE_VERSION = ""


/***********************************************************************/
/*          Science instrument name                         */
/*              -- Must be a string, max 20 len characters       */
/*          Example                                       */
/*              INSTRUMENT  = "NMC"                          */
/***********************************************************************/

INSTRUMENT = ""


/***********************************************************************/
/*          Spacecraft platform name                        */
/*              -- Must be a string, max len 25 characters       */
/*          Example                                       */
/*              PLATFORM  = "NOAA9"                          */
/***********************************************************************/

PLATFORM = ""
```

```
/***********************************************************************/
/*          ESDT description                                    */
/*              -- Must be a string, max len 60 characters         */
/*          Example                                            */
/*           DATA_TYPE_DESCRIPTION  = "NMC 12-hour forecast"          */
/***********************************************************************/

DATA_TYPE_DESCRIPTION = ""

/***********************************************************************/
/*          ESDT data provider (DAAC name to which files are delivered) */
/*              -- Must be a string, max len 50 characters         */
/*          Example                                            */
/*           PROVIDER  = "National Meteorological Center"          */
/***********************************************************************/

PROVIDER = ""

/***********************************************************************/
/*          Nominal ESDT file size in MB                         */
/*              -- Must be a floating point number (i.e., include a ".") */
/*              -- Must be greater than 0.000001                    */
/*          Example                                            */
/*           NOMINAL_SIZE = 1.5                                 */
/***********************************************************************/

NOMINAL_SIZE =

/***********************************************************************/
/*          Processing Level                                    */
/*              -- A string defining the level of processing for this   */
/*           ESDT.                                             */
/*             -- Must be a string of no more than 6 characters.    */
/*          Example                                            */
/*           PROCESSING_LEVEL  = "L0"                           */
/***********************************************************************/

PROCESSING_LEVEL = ""

/***********************************************************************/
/*          HDF Data Flag                                       */
/*            Informs DPS that the data within this ESDT will be     */
/*            HDF data (if set to Y).  Needed for DPS to correctly   */
/*            set the PCF entries for metadata access.              */
/*            -- A character value of either "Y" (YES) or "N" (NO).   */
/*           -- This will tell the Toolkit whether to get the      */
/*            metadata information from the HDF file of the ASCII    */
/*            metadata file.                                    */
/*          Example                                            */
/*           HDF_DATA = "N"                                     */
/***********************************************************************/

HDF_DATA = ""

/***********************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR files in the INPUT sections     */
/* of the PCF (PRODUCT, SUPPORT or INTERMEDIATE)                   */
/* (ignored for output files, which are always type "I")          */
/*                                                             */
```

```
/*              Dynamic flag -- flags whether an ESDT is dynamic       */
/*                 -- Allowed values:                              */
/*                    "S" -- Static file                          */
/*                    "I" -- Dynamic internal file               */
/*                    "E" -- Dynamic external file               */
/*                    "T" -- Interim/Intermediate file            */
/*          Example                                        */
/*             DYNAMIC_FLAG  = "E"                          */
/*********************************************************************/

DYNAMIC_FLAG = ""

/*********************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR Interim/Intermediate files */
/* (DYNAMIC_FLAG = "T")                                    */
/*                                               */
/*          Long Duration of Interim/Intermediate files of the     */
/*          ESDT before they are be deleted (because no longer needed).  */
/*             -- Must be a positive number (0 is NOT allowed).    */
/*           -- Time is specified in MINUTES.                  */
/*             -- This value should be long enough such that there is  */
/*                no chance that the file will be needed at the end of  */
/*                this duration.                          */
/*          Example                                        */
/*             INTERIM_LONG_DURATION  = 7200 (5 days)          */
/*********************************************************************/

INTERIM_LONG_DURATION  =

/*********************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR Interim/Intermediate files */
/* (DYNAMIC_FLAG = "T")                                    */
/*                                               */
/*          Short Duration of Interim/Intermediate files of the    */
/*          ESDT before they are be deleted (because no longer needed).  */
/*             -- Must be greater than or equal to 0.  It should only  */
/*                0 if no other PGE uses this Interim file (i.e. an */
/*                Interim file that a PGE uses internally between   */
/*                Processes).                          */
/*           -- Time is specified in MINUTES.                  */
/*             -- This value is a guess at the soonest (after use and  */
/*                any QA checks) at when the Interim File can be      */
/*                deleted.                             */
/*          Example                                        */
/*             INTERIM_SHORT_DURATION  = 1440 (24 Hours)         */
/*********************************************************************/

INTERIM_SHORT_DURATION  =

/*********************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR Dynamic Internal files       */
/* (DYNAMIC_FLAG = "I")                                    */
/*                                               */
/*          On Demand Deletion Interval.  This is the time between    */
/*          creation of a granule of this ESDT via an On Demand request  */
/*          and when this granule is deleted (because it has been  */
/*          distributed to the requestor).                  */
/*             -- Must contain a single P=V string, where          */
/*                P is one of { YEARS, MONTHS, THIRDS, WEEKS, DAYS,    */
```

```
/*                         HOURS, MINS, SECS, ORBITS}            */
/*           -- Must be greater than or equal to 1 week ("WEEKS=1").  */
/*            An error will be returned if the value specified   */
/*            is less than 1 week.                               */
/*          -- If not specified then the value defaults to 1 week    */
/*            ("WEEKS=1").                                 */
/*          Example                                       */
/*            ONDEMAND_DELETION_DURATION  = "WEEKS=1"            */
/**********************************************************************/

ONDEMAND_DELETION_DURATION = ""

/**********************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR Dynamic External files        */
/* (DYNAMIC_FLAG = "E")                                      */
/*                                                 */
/*          Data availability prediction method                */
/*            -- Must be one of {"ROUTINE", "NONROUTINE}         */
/*          -- "ROUTINE" = data is expected at regular intervals.    */
/*             "NONROUTINE" = data comes in sparatically.        */
/*                      No Period, Boundary or Duration is   */
/*                      required for NONROUTINE data.        */
/*          Example                                       */
/*            PREDICTION_METHOD  = "ROUTINE"                    */
/**********************************************************************/

PREDICTION_METHOD = ""

/**********************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR Dynamic External files        */
/* (DYNAMIC_FLAG = "E")                                      */
/*                                                 */
/*          Supplier name                                 */
/*            -- Must be a string, max len 30 characters        */
/*          Example                                       */
/*            SUPPLIER_NAME  = "NOAA"                          */
/**********************************************************************/

SUPPLIER_NAME = ""

/**********************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR Dynamic External files        */
/* (DYNAMIC_FLAG = "E")                                      */
/*                                                 */
/*          Nominal collection period within granule           */
/*          -- Must contain a single P=V string, where          */
/*             P is one of { YEARS, MONTHS, THIRDS, WEEKS, DAYS,    */
/*                      HOURS, MINS, SECS, ORBITS}           */
/*            -- NOTE that if ORBITS are used PROCESSING_BOUNDARY    */
/*             must be set to "START_OF_ORBIT".             */
/*             -- This value is ignored for PREDICTION_METHOD  =  */
/*             "NONROUTINE"                                */
/*          Example                                       */
/*            PERIOD  = "HOURS=12"                             */
/**********************************************************************/

PERIOD = ""

/**********************************************************************/
```

```
/* THIS PARAMETER IS ONLY REQUIRED FOR Dynamic External files        */
/* (DYNAMIC_FLAG = "E")                                        */
/*                                             */
/*          Nominal time boundary on which ESDT arrives        */
/*          -- Must contain 1 or more P=V strings, where P is one of   */
/*              { START_OF_HOUR, START_OF_6HOUR, START_OF_DAY,  */
/*                START_OF_WEEK, START_OF_ONE_THIRD_MONTH,        */
/*                START_OF_MONTH, START_OF_YEAR, START_DATE,       */
/*                START_OF_ORBIT };                    */
/*              also, "+<n>" or "-<n>"may be added to any of these,   */
/*              where <n> specifies integer seconds.           */
/*              For START_DATE an "=" can be added followed by the    */
/*              start date.                         */
/*            -- NOTE that START_OF_ORBIT must be used for Data based  */
/*              on an Orbit Model.  A file of named            */
/*              ORBIT_<platform>.odl must be present.            */
/*             -- This value is ignored for PREDICTION_METHOD  =  */
/*              "NONROUTINE"                          */
/*          Example                             */
/*            BOUNDARY   = "START_OF_DAY+10800"                  */
/************************************************************************/

BOUNDARY = ""

/************************************************************************/
/* OPTIONAL PARAMETER                                  */
/* ONLY USED FOR Dynamic External files                        */
/* (DYNAMIC_FLAG = "E")                                  */
/*                                             */
/*          Duration of the data.                        */
/*          -- Defines the length of time covered by the data.        */
/*          -- Only needed if length of time covered by the data     */
/*             differs from the value specified in PERIOD.        */
/*          -- Must contain a single P=V string, where            */
/*                P is one of { YEARS, MONTHS, THIRDS, WEEKS, DAYS,    */
/*                          HOURS, MINS, SECS, ORBITS}            */
/*            -- NOTE that if ORBITS are used PROCESSING_BOUNDARY     */
/*             must be set to "START_OF_ORBIT".                 */
/*            -- This value is ignored for PREDICTION_METHOD  =  */
/*              "NONROUTINE"                          */
/*          Example                             */
/*            DURATION = "HOURS=12"                       */
/************************************************************************/

DURATION = ""

/************************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR Dynamic External files        */
/* (DYNAMIC_FLAG = "E")                                  */
/*                                             */
/*          Avg delay between granule collection and arrival, in secs   */
/*            -- Must be a positive integer                 */
/*          Example                             */
/*            DELAY = 43200                           */
/************************************************************************/

DELAY =

/************************************************************************/
```

```
/*          Spatial characteristics of the Data Type.            */
/*              -- Must be a character, "Y" = Yes, spacial       */
/*                 characteristics exist, "N" = No, spacial      */
/*                 characteristics do not exist.                 */
/*          Example                                              */
/*              SPATIAL_FLAG  = "Y"                              */
/************************************************************************/

SPATIAL_FLAG  = ""

/************************************************************************/
/* OPTIONAL parameter                                          */
/*          Distinct Parameter for Granule naming                */
/*              -- A String, max length 80 characters.           */
/*            -- A value that will allow unqiue naming of granules  */
/*             produced by a PGE.                                */
/*            -- Must be the name of a metadata parameter defined in  */
/*             a METADATA_DEFINITION objected.  If a parameter is  */
/*             is specified for which no METADATA_DEFINITION object  */
/*             exists an error will be raised during ODL parsing.  */
/*            -- Supports what are called Multi-Granule ESDTs.  These  */
/*                are ESDTs that have multiple granules for the same  */
/*             time period where the only difference between the  */
/*             granules is metadata parameters.                 */
/*            -- NOTE that this parameter must be unqiue without  */
/*             including KEY_PARAMETER_NAME and KEY_PARAMETER_VALUE.  */
/*             If the parameter requires it, then they must still be  */
/*             specified, but the value specified for            */
/*             DISTINCT_PARAMETER cannot need them to be consided  */
/*             unique.                                         */
/*          Example                                              */
/*              DISTINCT_PARAMETER  = "CAMERA"                    */
/************************************************************************/

DISTINCT_PARAMETER  = ""

/************************************************************************/
/* Use object                                                  */
/*                                                             */
/* Defines the DAAC(s) where the data is used.                  */
/*                                                             */
/* There should be one of these for every DAAC where the data type is  */
/* used.  Delete or replicate this object as necessary.         */
/*                                                             */
/* This object is really only required for data that is used at a DAAC  */
/* other than where it's produced.                             */
/*                                                             */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present      */
/************************************************************************/

OBJECT = USE_OBJECT

/************************************************************************/
/*          Class (object counter, used only to distinguish objects)  */
/*              -- Must be an integer                            */
/*              -- Must be unique in this file                   */
/*          Example                                              */
/*              CLASS = 1                                        */
/************************************************************************/
```

```
   CLASS = 1

/*********************************************************************/
/*          DAAC where the Data Type is used.                   */
/*              -- Must be a string, max len 4 characters.  Use the    */
/*           DAAC abberviation (i.e. GSFC)                    */
/*              -- There should be one of these for every DAAC where   */
/*               the data type is used.                    */
/*          Example                             */
/*             USED_BY  = "GSFC"                         */
/*********************************************************************/

   USED_BY = ""

/*********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/*********************************************************************/

END_OBJECT = USE_OBJECT

/*********************************************************************/
/* THIS PARAMETER IS REQUIRED FOR ALL types of file but STATIC (S)    */
/* (DYNAMIC_FLAG = "S")                              */
/*                                  */
/*          DAAC where the Data Type is archived.               */
/*              -- Must be a string, max len 4 characters.  Use the    */
/*           DAAC abberviation (i.e. GSFC)                    */
/*          Example                             */
/*             ARCHIVED_AT  = "GSFC"                     */
/*********************************************************************/

ARCHIVED_AT  = ""

/*********************************************************************/
/* THIS PARAMETER IS ONLY REQUIRED FOR ALL types of file but STATIC (S) */
/* (DYNAMIC_FLAG = "S")                              */
/*                                  */
/*          DAAC where the Data Type is processed.              */
/*              -- Must be a string, max len 4 characters.  Use the    */
/*           DAAC abberviation (i.e. GSFC)                    */
/*          Example                             */
/*             PROCESSED_AT  = "GSFC"                    */
/*********************************************************************/

PROCESSED_AT  = ""

/*********************************************************************/
/* File Types Object                          */
/*                                  */
/* THIS OBJECT IS REQUIRED FOR all ESDTs that can have multiple files  */
/* per data granule.  It is NOT needed for ESDTs where each file  */
/* represents a single granule (those inputs in the PGE ODL file that  */
/* have "Single File Granule" for the File Type).           */
/*                                  */
/* It is up to the PGE writer to detemine if multiple files (whether   */
/* different types or multiple files for the same type) are        */
/* read/written by the PGE.  Files and granules differ because a  */
/* a granule is the smallest amount of data recognized by the system,  */
```

```
/* but one granule may be made up of several files.  These files  */
/* may be of different types, so that only specific information    */
/* (specific files) can be requested as input.                    */
/*                                                       */
/* Defines the types of files and their maximum numbers that can be  */
/* associated with this ESDT.                              */
/*                                                       */
/* There should be one of these for every File Type that can be    */
/* associated with this ESDT.                              */
/*                                                       */
/* Note that this does NOT need to be added for L0 data.  Though    */
/* such granules are multi-file, they are handled differently by   */
/* PDPS.  There does not need to be a FILETYPE object in the ESDT ODL  */
/* for L0 data.                                           */
/*                                                       */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present         */
/************************************************************************/


OBJECT = FILETYPE


/************************************************************************/
/*         Class (object counter, used only to distinguish objects)   */
/*            -- Must be an integer                              */
/*            -- Must be unique in this file                     */
/*         Example                                          */
/*            CLASS = 1                                     */
/************************************************************************/


   CLASS = 1


/************************************************************************/
/*         Name of File Type.                                */
/*            -- Must be a string, max len 40 characters.  Should    */
/*         be meaningful in that the name indicates what sort of     */
/*         data is stored within this file type.                */
/*            -- There should be one of these for every File Type that */
/*         can be associated with this ESDT.                    */
/*         Example                                          */
/*            FILETYPE_NAME  = "Instrument Band 7"              */
/************************************************************************/


   FILETYPE_NAME = ""


/************************************************************************/
/*         Maximum Number of Files under this Type.            */
/*            -- Must be an integer.                          */
/*          -- Indicates the maximum number of files for the     */
/*         specified File Type.                             */
/*            -- Must be less than 1000.                       */
/*         Example                                          */
/*            MAXIMUM_NUM_FILES  = 10                          */
/************************************************************************/


   MAXIMUM_NUM_FILES =


/************************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present         */
/************************************************************************/
```

```
END_OBJECT = FILETYPE

/***********************************************************************/
/* Metadata Definition Object                                      */
/*                                                           */
/* Metadata Defintion objects are required if there are to be Metadata */
/* Checks or Metadata Queries on this ESDT.  The object defines the    */
/* metadata parameters and their types on which checks or queries will */
/* or can be performed.                                         */
/*                                                           */
/* The actual values for the checks and/or queries are defined in the  */
/* PGE ODL file. All that needs to be defined in this ESDT ODL file is */
/* the computer data type of the value.  NOTE that there can be a      */
/* Metadata Definition object in the ESDT file and NO corresponding    */
/* Metadata Checks or Query object in the PGE ODL file.  But if there  */
/* is a Metadata Checks or Query object in the PGE ODL file, there MUST */
/* be a corresponding Metadata Defintion in the ESDT ODL file.         */
/*                                                           */
/* This object is optional (only needed if there are Metadata Checks   */
/* or Metadata Query objects in the corresponding PGE ODL file).  */
/* There may be many of these objects per ESDT file.               */
/*                                                           */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/***********************************************************************/

OBJECT = METADATA_DEFINITION

/***********************************************************************/
/*          Class (object counter, used only to distinguish objects)   */
/*              -- Must be an integer                           */
/*              -- Must be unique in this file                     */
/*          Example                                         */
/*             CLASS = 1                                    */
/***********************************************************************/

   CLASS = 1

/***********************************************************************/
/*          Parameter name for possible check or query           */
/*              -- Must be a string, max len 40 characters        */
/*              -- Must be identical to parm name read in PGE ODL file */
/*          Example                                         */
/*            PARM_NAME   = "tbd_parm_name"                 */
/***********************************************************************/

   PARM_NAME = ""

/***********************************************************************/
/*          Container name above the parameter to be checked/queried  */
/*                                                           */
/*              -- If not needed, should be set to "NONE".         */
/*            -- Must be filled in (correctly) if there is a container */
/*             object or a group surrounding the parameter specified */
/*             by PARM_NAME.  This is because Inspects on granules */
/*             can only be performed at the highest level         */
/*             object in the metadata tree.                    */
/*              -- Must be a string, max len 100 characters      */
/*            -- For Product Specific Attributes (PSAs) this must be    */
/*             set to "AdditionalAttibutes"                    */
```

```
/*              Example                                        */
/*              For metadata that looks as follows:            */
/*                 GROUP = SOME_GROUP_NAME                      */
/*                   OBJECT = OBJECT_CONTAINER                  */
/*                       CLASS = "1"                            */
/*                     OBJECT = PARAMETER_WE_ARE_QUERYING_ON       */
/*                         NUM_VAL          = 1                    */
/*                         VALUE          = "Value_we_want"          */
/*                     END_OBJECT = PARAMETER_WE_ARE_QUERYING_ON       */
/*                   END_OBJECT = OBJECT_CONTAINER                */
/*                 END_GROUP = SOME_GROUP_NAME                  */
/*                                                        */
/*         This parameter would be set as follows:            */
/*            CONTAINER_NAME   = "SOME_GROUP_NAME"             */
/************************************************************************/

    CONTAINER_NAME = ""


/************************************************************************/
/*          Type of parameter for check or query              */
/*             -- Must be one of {FLOAT,INT,STR}              */
/*          Example                                        */
/*             TYPE  = "INT"                                */
/************************************************************************/

    TYPE = ""


/************************************************************************/
/*  Optional Parameter.  Defaults to empty string if not specified.    */
/*                                                        */
/*          Name of metadata parameter which provides a key into a    */
/*       a multi-containered object.  Such an object is the           */
/*       MeasuredParameters group in the inventory metadata.          */
/*             -- Must be a string, max len 40 characters.     */
/*             -- Must be present in the ESDT ODL file for this ESDT.   */
/*             -- Is matched with KEY_PARAMETER_VALUE to determine      */
/*                 the entry in a mult-containered metadata group. */
/*            -- For Product Specific Attibutes (PSAs), this entry      */
/*             should NOT be specified.                         */
/*            -- Because an ESDT may be used by more than one PGE, it    */
/*             is possible to have more than one               */
/*             KEY_PARAMETER_NAME/KEY_PARAMETER_VALUE pair       */
/*             (in multiple METADATA_DEFINITION objects)        */
/*             per ESDT ODL File.  Any PGE ODL file may only have  */
/*             a single KEY_PARAMETER_NAME/KEY_PARAMETER_VALUE pair. */
/*            -- For Metadata Checks or Qeuries within the       */
/*             MeasuredParameters group this should be set to the  */
/*             metadata field called "ParameterName".          */
/*          Example:                                        */
/*             KEY_PARAMETER_NAME = "ParameterName"             */
/************************************************************************/

        KEY_PARAMETER_NAME = ""


/************************************************************************/
/*  Optional Parameter.  Must be preset if KEY_PARAMETER_NAME exists.   */
/*  Defaults to the empty string if not specified.             */
/*                                                        */
/*          Value of metadata parameter which provides a key into a    */
```

```
/*         a multi-containered object.  Such an object is the            */
/*         MeasuredParameters group in the inventory metadata.           */
/*             -- Must be a string, max len 80 characters.       */
/*             -- Must be present in the ESDT ODL file for this ESDT.    */
/*             -- Is matched with KEY_PARAMETER_NAME to determine        */
/*                the entry in a mult-containered metadata group. */
/*           -- For Product Specific Attibutes (PSAs), this entry        */
/*            should NOT be specified.                            */
/*           -- Because an ESDT may be used by more than one PGE, it     */
/*            is possible to have more than one                  */
/*            KEY_PARAMETER_NAME/KEY_PARAMETER_VALUE pair         */
/*            (in multiple METADATA_DEFINITION objects)           */
/*            per ESDT ODL File.  Any PGE ODL file may only have  */
/*            a single KEY_PARAMETER_NAME/KEY_PARAMETER_VALUE pair. */
/*           -- For Metadata Checks or Queries within the        */
/*            MeasuredParameters group this should be set to the  */
/*            desired value of the metadata field called          */
/*            "ParameterName".                                    */
/*          Example:                                              */
/*            KEY_PARAMETER_VALUE = "LandCoverage"                */
/***********************************************************************/

        KEY_PARAMETER_VALUE = ""


/***********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present      */
/***********************************************************************/

END_OBJECT = METADATA_DEFINITION


/***********************************************************************/
/* Metadata Definition object may be repeated as needed          */
/***********************************************************************/


/***********************************************************************/
/* Production Chain Object                                */
/*                                                        */
/* THIS OBJECT is only needed for those ESDTs that will be produced    */
/* by On Demand Production Requests (Production Requests that are */
/* generated as a result of a request for an On Demand Product).    */
/*                                                        */
/* The Production Chain object surrounds a list (in order) of the    */
/* PGEs needed to produce a granule of this ESDT.  There may be one    */
/* PGE in the list (if that PGE takes in DYnamic External data and    */
/* produces this ESDT), or a chain of PGEs (if PGE A produces an  */
/* ESDT that is input to PGE B which produces THIS ESDT).         */
/*                                                        */
/* The information contained within this object will only be used if    */
/* an On Demand Request is for an ESDT which must have another    */
/* ESDT produced for the PGE that is to create the Product.      */
/*                                                        */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present      */
/***********************************************************************/

OBJECT = PRODUCTION_CHAIN

/***********************************************************************/
/*         Class (object counter, used only to distinguish objects)   */
```

```
/*              -- Must be an integer                                */
/*              -- Must be unique in this file                       */
/*          Example                                          */
/*            CLASS = 1                                      */
/**********************************************************************/

   CLASS = 1


/**********************************************************************/
/* PGE In Chain Object                                       */
/*                                                    */
/* THIS OBJECT defines a PGE that is part of the Production Chain    */
/* used to produce this ESDT.                                 */
/*                                                    */
/* The Production Chain object surrounds a list (in order) of the    */
/* PGEs needed to produce a granule of this ESDT.  There may be one  */
/* PGE in the list (if that PGE takes in DYnamic External data and   */
/* produces this ESDT), or a chain of PGEs (if PGE A produces an  */
/* ESDT that is input to PGE B which produces THIS ESDT).            */
/*                                                    */
/* The PGE_IN_CHAIN objects within the PRODUCTION_CHAIN object define  */
/* the PGEs (in order) that need to be run to produce this ESDT.  */
/* Only the PGE Name and Version are needed to identify the PGE, the  */
/* Profile Id will be the one with the DEFEAUL_PROFILE flag set.  */
/*                                                    */
/* The information contained within this object will only be used if   */
/* an On Demand Request is for an ESDT which must have another       */
/* ESDT produced for the PGE that is to create the Product.        */
/*                                                    */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present        */
/**********************************************************************/

OBJECT = PGE_IN_CHAIN

/**********************************************************************/
/*          Class (object counter)                            */
/*              -- Must be an integer                          */
/*              -- Must be unique in this file                 */
/*            -- Is used in this case to determine the order of the   */
/*                  PGEs.  CLASS = 1 is the first PGE in the chain.   */
/*          Example                                          */
/*            CLASS = 1                                      */
/**********************************************************************/

   CLASS = 1

/**********************************************************************/
/*          PGE name                                          */
/*              -- Must be a string, max len 10 characters      */
/*              -- The is the name of the PGE that makes up one entry   */
/*                  in the chain of PGEs.                       */
/*          Example                                          */
/*            PGE_NAME = "ssit"                               */
/**********************************************************************/

   PGE_NAME = ""


/**********************************************************************/
/*          PGE version                                       */
```

```
/*              -- Must be a string, max len 5 characters           */
/*              -- This is the version of the PGE that makes up one     */
/*                 entry in the chain of PGEs.                          */
/*          Example                                               */
/*              PGE_VERSION = "1.0"                                 */
/************************************************************************/

  PGE_VERSION = ""


/************************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/************************************************************************/

END_OBJECT = PGE_IN_CHAIN

/************************************************************************/
/* Repeat PGE_IN_CHAIN objects as needed to make up the Production    */
/* chain.                                                       */
/************************************************************************/


/************************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/************************************************************************/

END_OBJECT = PRODUCTION_CHAIN

/************************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED                       */
/************************************************************************/

END
```

A.1.3 ORBIT_ODL.template

```
/************************************************************************/
/*                                                          */
/*          TEMPLATE ORBIT MODEL METADATA ODL FILE           */
/*                                                          */
/*                                                          */
/* The SSIT operator's responsibility is to copy this file over and   */
/* edit it to add all necessary PDPS metadata values.            */
/*                                                          */
/* All ORBIT MODEL ODL files must reside in directory            */
/* $DPAT_RULE_SCIENCE_MD (set in the configruation files).        */
/*                                                          */
/* The operator must add a value to the right of the "=" for each    */
/* parameter.                                                 */
/*                                                          */
/* This file is only needed if the PGE has a period/boundary relating */
/* to orbit.                                                  */
/*                                                          */
/* There can be one or more ORBIT_MODEL objects defined in         */
/* this file so that multiple orbits can be defined for the same     */
/* platform.                                                  */
/*                                                          */
/* CHANGE LOG                                                 */
/*     -- Added Orbit_Path_Number.                    11/18/97  */
/*     -- Changed acceptable Date Format.             01/05/98  */
```

```
/*     -- Added another acceptable date format.       06/24/98  */
/*     -- Updated length of PLATFORM.                 08/13/98  */
/*     -- Fixed where this file is located in above   10/01/98  */
/*         comments.                                            */
/**********************************************************************/


/**********************************************************************/
/*          Spacecraft platform name for the Orbit Model.       */
/*              -- Must be a string, max len 25 characters      */
/*          Example                                      */
/*            PLATFORM = "TRMM"                            */
/**********************************************************************/

PLATFORM = ""

/**********************************************************************/
/* Orbit Model object                                      */
/*                                                */
/* Defines the Orbit Model for a single orbit                    */
/*                                                */
/* Replicate for the defining of multiple orbits for the same platform. */
/*                                                */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present      */
/**********************************************************************/

OBJECT = ORBIT_MODEL

/**********************************************************************/
/*          Class (object counter, used only to distinguish objects)  */
/*              -- Must be an integer                    */
/*              -- Must be unique in this file for this type of object  */
/*              -- Must be greater than 0.                   */
/*          Example                                      */
/*            CLASS = 1                                 */
/**********************************************************************/

     CLASS = 1

/**********************************************************************/
/*        Number of the Orbit                          */
/*            -- Must be an integer                       */
/*            -- Must be >=  0                            */
/*        Example                                       */
/*          ORBIT_NUMBER = 12                             */
/**********************************************************************/

     ORBIT_NUMBER =

/**********************************************************************/
/*        Path Number of the Orbit                         */
/*            -- Must be an integer                       */
/*            -- Must be >= 0 and <= 233                 */
/*        Example                                       */
/*          ORBIT_PATH_NUMBER = 3                         */
/**********************************************************************/

     ORBIT_PATH_NUMBER =

/**********************************************************************/
```

```
/*          The period of the orbit (a duration).                */
/*              -- Must contain a single P=V string, where        */
/*                 P is one of { MONTHS, WEEKS, DAYS, HOURS, MINS, SECS} */
/*          Example                                               */
/*              ORBIT_PERIOD = "HOURS=100"                        */
/*****************************************************************/

      ORBIT_PERIOD = " "


/*****************************************************************/
/*          The starting date/time of the orbit.                  */
/*              -- Must contain the date and time of the orbit.    */
/*              -- The format for the date/time string can be one of the */
/*          follwoing:                                            */
/*              "MMM DD YYYY HH:MM:SS", where                     */
/*               YYYY=4 digit year, MMM=3 character abreviation for    */
/*          Month, DD=2 digit Day, HH=Hours, MM=Minutes,          */
/*          SS=Seconds.  The time is accepted as UTC.             */
/*                                                                */
/*              "MM/DD/YYYY HH:MM:SS"                             */
/*               YYYY=4 digit year, MM=2 digit Month,             */
/*          DD=2 digit Day, HH=Hours, MM=Minutes,                 */
/*          SS=Seconds.  The time is accepted as UTC.             */
/*            -- NOTE that the format for the date of MM/DD/YY will    */
/*          no longer be accepted because it did not handle years     */
/*          after 1999 correctly.                                 */
/*          Example                                               */
/*              ORBIT_START = "Oct 31 1996 22:01:55"              */
/*****************************************************************/

      ORBIT_START = " "


/*****************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present      */
/*****************************************************************/


END_OBJECT = ORBIT_MODEL


/*****************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED                       */
/*****************************************************************/


END
```

A.1.4 TILE_ODL.template

```
/*****************************************************************/
/*                                                              */
/*          TEMPLATE TILE DEFINITION METADATA ODL FILE          */
/*                                                              */
/*                                                              */
/* The SSIT operator's responsibility is to copy this file over and   */
/* edit it to add all necessary PDPS metadata values.          */
/*                                                              */
/* Each Tile Scheme used by a PGE must have a corresponding TILE  */
/* DEFINITION metadata ODL file.                                */
/*                                                              */
/* All TILE DEFINITION ODL files must reside in directory         */
/* $DPAT_RULE_SCIENCE_MD.  Each must be named TILE_<tile scheme>.odl   */
/*                                                              */
```

```
/* For a PGE to use a tile scheme, it must have SCHEDULE_TYPE =   */
/* "Tile".  TILE_SCHEME_TYPE must equal the tiling scheme to be used.   */
/*                                                         */
/* The operator must add a value to the right of the "=" for each      */
/* parameter.                                               */
/*                                                         */
/*                                                   */
/* CHANGE LOG                                            */
/*    -- Removed the concept of CLUSTERs.                  01/18/98  */
/*        Added COORDINATE object.                          */
/*    -- Updated various descriptions to make them better.  02/04/98   */
/*                                                         */
/*                                                         */
/***********************************************************************/


/***********************************************************************/
/*        Tile Scheme                                          */
/*             -- Must be a string, max len 20 characters      */
/*           -- There can be NO spaces in the string.          */
/*             -- Tile Scheme must be identical to             */
/*                 Tile Scheme used as part of ODL filename,       */
/*                 which in turn was generated from the        */
/*                 TILE_SCHEME_NAME in the PGE ODL file.       */
/*        Example                                        */
/*           TILE_SCHEME_NAME  = "Earth_Squared"              */
/***********************************************************************/

TILE_SCHEME_NAME = " "

/***********************************************************************/
/* Tile object                                              */
/*                                                   */
/* Defines a tile for the scheme defined by TILE_SCHEME_NAME.       */
/* Each tile must be defined seperately, with an ID, and associated    */
/* coordinates.                                             */
/*                                                   */
/* There should be a Tile object for every tile in the Tiling Scheme.  */
/*                                                   */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present       */
/***********************************************************************/

   OBJECT = TILE

/***********************************************************************/
/*        Class (object counter, used only to distinguish objects)   */
/*            -- Must be an integer                         */
/*            -- Must be unique in this file for this type of object  */
/*            -- Must be greater than 0.                      */
/*        Example                                        */
/*           CLASS = 1                                   */
/***********************************************************************/

   CLASS = 1

/***********************************************************************/
/*        ID of Tile                                      */
/*             -- Must be an integer                        */
/*             -- Must greater than 0 but less than max integer.  */
/*          -- Tiles should be listed sequentially (though no    */
```

```
/*              checking for this is done by software).           */
/*              -- This must be unique throughout the system.  This      */
/*              means that if this Tile Id is defined in other Tile   */
/*          Schemes, it must have the same coordinates and      */
/*              description.                                     */
/*          Example                                      */
/*              TILE_ID = 12                                 */
/**********************************************************************/

   TILE_ID =

/**********************************************************************/
/*          Description of a Tile                          */
/*              -- A String of characters, max 255.           */
/*              -- Describes what the Tile is for, perhaps its       */
/*               geographic location or area that it covers.     */
/*          Example                                      */
/*              TILE_DESCRIPTION = "Upper North America"            */
/**********************************************************************/

   TILE_DESCRIPTION = ""

/**********************************************************************/
/* Tile Coordinate object                                  */
/*                                                  */
/* Defines a coordinate (Latitude and Longitude) for a tile.         */
/*                                                  */
/* Each tile must have at least 4 TILE_COORDINATE objects defined. More */
/* (than 4) such objects are permitted to better define the tile. */
/*                                                  */
/* Cooridnate objects must follow a clockwise sequence such that if    */
/* lines were drawn between the points in the order they are given the  */
/* desired shape would be drawn.                             */
/*
/*
/*
/*
/*
/* For example:                                            */
/*          Coordinate 1                    Coordinate 2            */
/*              o-------------------------------->o              */
/*              ^                                |              */
/*              |                                |              */
/*              |                                v              */
/*              o<--------------------------------o              */
/*          Cooridnate 4                    Coordinate 3            */
/*                                                  */
/*    Or:                                                  */
/*
/*          Coordinate 2                    Coordinate 3            */
/*              o-------------------------------->o              */
/*              ^                                |              */
/*              |                                |              */
/*              |                                v              */
/*              o<--------------------------------o              */
/*          Cooridnate 1                    Coordinate 4            */
/*                                                  */
/*                                                  */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present        */
```

```
/*********************************************************************/


   OBJECT = TILE_COORDINATE

/*********************************************************************/
/*          Class (object counter, used only to distinguish objects)   */
/*              -- Must be an integer                                  */
/*              -- Must be unique in this file for this type of object */
/*              -- Must be greater than 0.                             */
/*          Example                                              */
/*              CLASS = 1                                        */
/*********************************************************************/

   CLASS = 1

/*********************************************************************/
/*          Latitude Coordinate                                  */
/*              -- Must be one per Coordinate object.            */
/*                -- Must be an float                            */
/*          Example                                              */
/*              LATITUDE = 12.15                                 */
/*********************************************************************/

   LATITUDE =

/*********************************************************************/
/*          Longitude Coordinate                                 */
/*              -- Must be one per Coordinate object.            */
/*                -- Must be an float                            */
/*          Example                                              */
/*              LONGITUDE = -43.22                               */
/*********************************************************************/

   LONGITUDE =

/*********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present        */
/*********************************************************************/

   END_OBJECT = TILE_COORDINATE

/*********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present        */
/*********************************************************************/

   END_OBJECT = TILE

/*********************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED                         */
/*********************************************************************/

END
```

A.1.5 PATHMAP_ODL.template

```
/*********************************************************************/
/*                                                               */
/*          TEMPLATE PATHMAP DEFINITION METADATA ODL FILE          */
```

```
/*                                                                    */
/*                                                                    */
/* The SSIT operator's responsibility is to copy this file over and   */
/* edit it to add all necessary PATH MAP metadata values.             */
/*                                                                    */
/* A PATHMAP defines the mapping between Absolute Path Number          */
/* a sequential numbering from 1-233 and Mapped Path Number which      */
/* is the intrepeted 1-233 number.                                     */
/*                                                                    */
/* If a PGE defines a PATHMAP in the PGE ODL then there must be a       */
/* matching PATHMAP DEFINITION metadata ODL file and the PGE must have  */
/* a SCHEDULE_TYPE = "Orbit".                                           */
/*                                                                    */
/* All PATHMAP DEFINITION ODL files must reside in directory            */
/* $DPAT_RULE_SCIENCE_MD.  Each must be named                          */
/* PATHMAP_<Pathmap_Name>.odl.   Note there can be NO spaces in the    */
/* Pathmap_Name because it is used as a filename.                      */
/*                                                                    */
/* For a PGE to use a PATHMAP, the PATHMAP_NAME parameter in the PGE    */
/* ODL file must equal the Pathmap_Name to be used.                    */
/*                                                                    */
/* The operator must add a value to the right of the "=" for each      */
/* parameter.                                                          */
/*                                                                    */
/*                                                                    */
/* CHANGE LOG                                                          */
/*                                                                    */
/*                                                                    */
/*********************************************************************/


/*********************************************************************/
/*          Spacecraft platform name for the Orbit Model.            */
/*              -- Must be a string, max len 20 characters           */
/*          Example                                                  */
/*             PLATFORM = "TRMM"                                      */
/*********************************************************************/

PLATFORM = ""

/*********************************************************************/
/*          Pathmap Name                                             */
/*              -- Must be a string, max len 25 characters           */
/*            -- There can be NO spaces in the string.               */
/*              -- Pathmap Name must be identical to                 */
/*                   Pathmap Name used as part of ODL filename,       */
/*                   which in turn was generated from the            */
/*                   PATHMAP_NAME in the PGE ODL file.               */
/*          Example                                                  */
/*             PATHMAP_NAME  = "Some_Pathmap"                         */
/*********************************************************************/

PATHMAP_NAME = " "

/*********************************************************************/
/* Pathmap Entry Object                                              */
/*                                                                    */
/* Defines a mapping between Absolute Path Number                     */
/* a sequential numbering from 1-233 and Mapped Path Number which     */
/* is the intrepeted 1-233 number.                                    */
```

```
/*                                                                          */
/* There should be a Pathmap Entry object for each 1-233 Path Number.       */
/* An error will be returned if one of the path numbers is missed.          */
/*                                                                          */
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present                 */
/****************************************************************************/

   OBJECT = PATHMAP_ENTRY

/****************************************************************************/
/*          Class (object counter, used only to distinguish objects)    */
/*              -- Must be an integer                                       */
/*              -- Must be unique in this file for this type of object   */
/*              -- Must be greater than 0.                                  */
/*          Example                                                         */
/*             CLASS = 1                                                    */
/****************************************************************************/

      CLASS = 1

/****************************************************************************/
/*          Absolute Path Number                                            */
/*              -- Must be an integer                                       */
/*              -- Must be between 1-233.                                    */
/*          Example                                                         */
/*             ABSOLUTE_PATH = 20                                           */
/****************************************************************************/

      ABSOLUTE_PATH =

/****************************************************************************/
/*          Mapped Path Number                                              */
/*              -- Must be an integer.                                      */
/*              -- Must be between 1-233.                                    */
/*          Example                                                         */
/*             MAPPED_PATH = 27                                             */
/****************************************************************************/

      MAPPED_PATH = ""


/****************************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED if it is present                 */
/****************************************************************************/

   END_OBJECT = PATHMAP_ENTRY

/****************************************************************************/
/* THE FOLLOWING LINE MUST NOT BE MODIFIED                                  */
/****************************************************************************/
END
```

A.2   Typical ASTER PGE & ESDT ODL Files

Listings are provided for the following ASTER ODL files:

A.2.1 ASTER PGE ODL file for PGE_NAME BTS
A.2.2 ASTER ESDT ODL file for DATA_TYPE_NAME AST_LIB
A.2.3 ASTER ESDT ODL file for DATA_TYPE_NAME AST_ANC
A.2.4 ASTER ESDT ODL file for DATA_TYPE_NAME AST_04
A.2.5 ASTER ESDT ODL file for DATA_TYPE_NAME AST_09T

AST_LIB, AST_ANC and AST_04 are referenced within the PGE .

A typical ASTER PGE will differ from the example here by the PGE_NAME, the
specific input/output files referenced, and runtime parameters. However, the
overall structure of a given ASTER PGE ODL file would be similar to the one
used here. (N.B. The ODL files shown here are associated with the ASTER
version v2.2.34 software)

A.2.1 ASTER PGE BTS ODL

```
PGE_NAME = "BTS"
PGE_VERSION = "2.2h"
PROFILE_ID = 1
PROFILE_DESCRIPTION = "ASTER Brightness Temp with QA"
PLATFORM = "AM1"
INSTRUMENT = "ASTER"
MINIMUM_OUTPUTS = 1
SCHEDULE_TYPE = "Data"
PROCESSING_PERIOD = "SECS=1"
PROCESSING_BOUNDARY = "START_OF_SEC"
PGE_SSW_VERSION = "2.2h"

OBJECT = PCF_ENTRY
   CLASS = 11
   LOGICAL_ID = 15004
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "AST_L1B"
   DATA_TYPE_VERSION = "001"
   DATA_TYPE_REQUIREMENT = 1
   INPUT_TYPE = "Required"
   KEY_INPUT = "Y"
   NUMBER_NEEDED = 1
/**** Entry needed for all I/O (except for Temporary) ****/
/**** Only modify if multiple files and/or file types for this PCF entry ****/
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*Bright-Temp-LUT-v3.hdf*/
OBJECT = PCF_ENTRY
   CLASS = 12
   LOGICAL_ID = 15330
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "AST_ANC"
   DATA_TYPE_VERSION = "001"
   DATA_TYPE_REQUIREMENT = 1
   SCIENCE_GROUP = L1
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
```

```
/**** Entry needed for all I/O (except for Temporary) ****/
/**** Only modify if multiple files and/or file types for this PCF entry ****/
    OBJECT = FILETYPE
        FILETYPE_NAME = "Single File Granule"
        CLASS = 1
    END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
/**** "atmcorr-v3-dec.hdf" ****/
    CLASS = 29
    LOGICAL_ID = 15152
    PCF_FILE_TYPE = 3
    DATA_TYPE_NAME = "AST_ANC"
    DATA_TYPE_VERSION = "001"
    DATA_TYPE_REQUIREMENT = 1
    SCIENCE_GROUP = "L2"
    INPUT_TYPE = "Required"
    NUMBER_NEEDED = 1
/**** Entry needed for all I/O (except for Temporary) ****/
/**** Only modify if multiple files and/or file types for this PCF entry ****/
    OBJECT = FILETYPE
        FILETYPE_NAME = "Single File Granule"
        CLASS = 1
    END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
/**** "FBA_Filter_File_1.cal" ****/
    CLASS = 30
    LOGICAL_ID = 15151
    PCF_FILE_TYPE = 3
    DATA_TYPE_NAME = "AST_ANC"
    DATA_TYPE_VERSION = "001"
    DATA_TYPE_REQUIREMENT = 1
    SCIENCE_GROUP = "O30"
    INPUT_TYPE = "Required"
    NUMBER_NEEDED = 1
/**** Entry needed for all I/O (except for Temporary) ****/
/**** Only modify if multiple files and/or file types for this PCF entry ****/
    OBJECT = FILETYPE
        FILETYPE_NAME = "Single File Granule"
        CLASS = 1
    END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

/* QA2-binning-intervals-v1.cal */
OBJECT = PCF_ENTRY
    CLASS = 13
    LOGICAL_ID = 15913
    PCF_FILE_TYPE = 1
    DATA_TYPE_NAME = "AST_ANC"
    DATA_TYPE_VERSION = "001"
    DATA_TYPE_REQUIREMENT = 1
    SCIENCE_GROUP = O98
    INPUT_TYPE = "Required"
    NUMBER_NEEDED = 1
/**** Entry needed for all I/O (except for Temporary) ****/
/**** Only modify if multiple files and/or file types for this PCF entry ****/
```

```
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY

   /**/
   /* QA_thresholds.dat */
   OBJECT = PCF_ENTRY
      CLASS = 14
      LOGICAL_ID = 15120
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "AST_ANC"
      DATA_TYPE_VERSION = "001"
      DATA_TYPE_REQUIREMENT = 1
      SCIENCE_GROUP = O97
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
   /**** Entry needed for all I/O (except for Temporary) ****/
   /**** Only modify if multiple files and/or file types for this PCF entry ****/
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY

   /*Output Product*/
   OBJECT = PCF_ENTRY
      CLASS = 15
      LOGICAL_ID = 15010
      PCF_FILE_TYPE = 2
      DATA_TYPE_NAME = "AST_04"
      DATA_TYPE_VERSION = "001"
      YIELD = 1
      ASSOCIATED_MCF_ID = 15114
      SCIENCE_GROUP = "S1"
      INSTANCE = 0
      MINIMUM_SIZE = 5
      MAXIMUM_SIZE = 10
   /**** Entry needed for all I/O (except for Temporary) ****/
   /**** Only modify if multiple files and/or file types for this PCF entry ****/
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY

   OBJECT = PCF_ENTRY
      CLASS = 126
      LOGICAL_ID = 15015
      PCF_FILE_TYPE = 2
      DATA_TYPE_NAME = "ASTALGRN"
      DATA_TYPE_VERSION = "001"
      YIELD = 1
      SCIENCE_GROUP = "S3"
      ASSOCIATED_MCF_ID = 15119
      INSTANCE = 0
      MINIMUM_SIZE = 0
      MAXIMUM_SIZE = 0
```

```
/**** Entry needed for all I/O (except for Temporary) ****/
/**** Only modify if multiple files and/or file types for this PCF entry ****/
    OBJECT = FILETYPE
        FILETYPE_NAME = "Single File Granule"
        CLASS = 1
    END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
    CLASS = 16
    LOGICAL_ID = 15602
    PCF_FILE_TYPE = 5
    PGE_PARAMETER_NAME = "PGE Major Version"
    PGE_PARAMETER_DEFAULT = "2"
    PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
    CLASS = 17
    LOGICAL_ID = 15603
    PCF_FILE_TYPE = 5
    PGE_PARAMETER_NAME = "PGE Minor Version"
    PGE_PARAMETER_DEFAULT = "2"
    PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
    CLASS = 18
    LOGICAL_ID = 16200
    PCF_FILE_TYPE = 5
    PGE_PARAMETER_NAME = "QA PGE Major Version"
    PGE_PARAMETER_DEFAULT = "2"
    PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
    CLASS = 19
    LOGICAL_ID = 16201
    PCF_FILE_TYPE = 5
    PGE_PARAMETER_NAME = "QA PGE Minor Version"
    PGE_PARAMETER_DEFAULT = "2"
    PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
    CLASS = 20
    LOGICAL_ID = 15604
    PCF_FILE_TYPE = 5
    PGE_PARAMETER_NAME = "Brightness Temperature LUT"
    PGE_PARAMETER_DEFAULT = "3"
    PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
    CLASS = 139
    LOGICAL_ID = 15167
    PCF_FILE_TYPE = 5
    PGE_PARAMETER_NAME = "BrtTmp Lookup Table Version"
    PGE_PARAMETER_DEFAULT = "3"
```

```
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 22
   LOGICAL_ID = 15165
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Atmos Corr. LUT Version"
   PGE_PARAMETER_DEFAULT = "3"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 24
   LOGICAL_ID = 15166
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "FBA Filters File Version"
   PGE_PARAMETER_DEFAULT = "3"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 21
   LOGICAL_ID = 15914
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "QA2 Binning Interval Version"
   PGE_PARAMETER_DEFAULT = "1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 135
   LOGICAL_ID = 15320
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Alert File indirection"
   PGE_PARAMETER_DEFAULT = "15015:1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 135
   LOGICAL_ID = 15321
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "PGE Name"
   PGE_PARAMETER_DEFAULT = "Brightness Temperature at the Sensor"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

END
```

A.2.2 ASTER ESDT AST_LIB ODL

```
DATA_TYPE_NAME = "AST_L1B"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "ASTER"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "ASTER Level 1B Data Set Registered Radiance at the
                         Sensor"
PROVIDER = "EROS Data Center"
```

```
NOMINAL_SIZE = 120.0
PROCESSING_LEVEL = "L1"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "NONROUTINE"
SUPPLIER_NAME = "EDC"
PERIOD = "SECS=1"
DURATION = "SECS=1"
BOUNDARY = "START_OF_SEC"
DELAY = 1
SPATIAL_FLAG  = "Y"
ARCHIVED_AT  = "EDC"
PROCESSED_AT  = "EDC"


END
```

A.2.3 ASTER ESDT AST_ANC ODL

```
DATA_TYPE_NAME = "AST_ANC"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "ASTER"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "HDF Ancillary data for ASTER"
PROVIDER = "Goddard Space Flight Center"
PROCESSING_LEVEL = "L1"
HDF_DATA = "Y"
NOMINAL_SIZE = 1.0
DYNAMIC_FLAG = "S"


END
```

A.2.4 ASTER ESDT AST_04 ODL

```
DATA_TYPE_NAME = "AST_04"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "ASTER"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "ASTER Level 2 Brightness Temperature at the Sensor"
PROVIDER = "Goddard Space Flight Center"
PROCESSING_LEVEL = "L1"
HDF_DATA = "Y"
NOMINAL_SIZE = 4.744895
DYNAMIC_FLAG = "I"
ARCHIVED_AT  = "EDC"
PROCESSED_AT  = "EDC"


END
```

A.2.5 ASTER ESDT AST_09T ODL

```
DATA_TYPE_NAME = "AST_09T"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "ASTER"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "ASTER Level 2 Surface Radiance Product (TIR)"
```

```
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 9.439935
PROCESSING_LEVEL = "L1"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "EDC"
PERIOD = "SECS=1"
BOUNDARY = "START_OF_SEC"
DELAY = 1
SPATIAL_FLAG  = "N"
ARCHIVED_AT  = "EDC"
PROCESSED_AT  = "EDC"

END
```

A.3   Typical MISR PGE & ESDT ODL Files

Listings are provided for the following MISR ODL files:

A.3.1 MISR PGE ODL file for PGE_NAME MPGE1 (M1AN)

A.3.2 MISR ESDT MISANCGM ODL

A.3.3 MISR ESDT MIRCCT ODL

A.3.4 MISR ESDT MISL0AN ODL

A.3.5 MISR ESDT ActSched ODL

A.3.6 MISR ESDT MIANCSSC ODL

A.3.7 MISR ESDT MIANCAGP ODL

A.3.8 MISR ESDT MIANPPAN ODL

A.3.9 MISR ESDT MISL0SY1 ODL

A.3.10 MISR ESDT MISL0SY2 ODL

A.3.11 MISR ESDT MISL0SY3 ODL

A.3.12 MISR ESDT MIRFOIAN ODL

A.3.13 MISR ESDT MIB2GEOP ODL

A.3.14 MISR ESDT MIANCARP ODL (Version# 001)

A.3.15 MISR ESDT MIANCARP ODL (Version# 002)

A.3.16 MISR ESDT MICNFG ODL

A.3.17 MISR ESDT AM1EPHN0 ODL

A.3.18 MISR ESDT AM1ATTNF ODL

A.3.19 MISR ESDT MIANRCCH ODL

A.3.20 MISR ESDT MIL1A ODL

A.3.21 MISR ESDT MISBR ODL

A.3.22 MISR ESDT MISQA ODL

A.3.23 MISR ESDT MI1B2T ODL

A.3.24 MISR ESDT MI1B2E ODL

A.3.25 MISR ESDT MIRCCM ODL

A.3.26 MISR ESDT MI1B1 ODL

A.3.27 MISR ESDT MIB1LM ODL

A typical MISR PGE will differ from the example here by the PGE_NAME, the specific input/output files referenced, and runtime parameters. However, the overall structure of a given MISR PGE ODL file would be similar to the one used here. (N.B. The ODL files shown here are associated with the MISR version v2.1.3 Patch 2 software.)

A.3.1 MISR PGE MPGE1AN ODL *(profile #1)*

```
PGE_NAME = "M1AN"
PGE_VERSION = "21302"
PROFILE_ID = 1
PROFILE_DESCRIPTION = "MISR PGE 1 AN - Version V21302, SSI&T 17 March 2001"
PLATFORM = "AM1"
INSTRUMENT = "MISR"

/* MISR PGE 1 produces at a minimum 11 output files including QA        */
MINIMUM_OUTPUTS = 11


SCHEDULE_TYPE = "Orbit"
PROCESSING_PERIOD = "ORBITS=1"
PROCESSING_BOUNDARY = "START_OF_ORBIT"
PATHMAP_NAME = "MISR"

/*  PGE_SSW_VERSION should match the PGE_VERSION                        */
PGE_SSW_VERSION = "21302"


OBJECT = EXIT_MESSAGE
   CLASS= 1
   EXIT_CODE = 0
   EXIT_MESSAGE = "CODE(0):  Successful Completion of MISR PGE 1 AN"
END_OBJECT = EXIT_MESSAGE
OBJECT = EXIT_MESSAGE
   CLASS= 2
   EXIT_CODE = 202
   EXIT_MESSAGE = "CODE(202):  Execution Failure of MISR PGE 1 AN"
END_OBJECT = EXIT_MESSAGE

/*********************************************************************/
```

```
/*                    MISR PGE 1 AN   Inputs                              */
/*     Inputs:                                                            */
/*     LID        ESDT.Version       Science Group                        */
/************************************************************************/
/*                    MISR PGE 1 AN Inputs                                */
/*     Inputs:                                                            */
/*     LID        ESDT.Version       Science Group                        */
/*     190        MISANCGM.002       Dynamic External Input               */
/*     227        MIRCCT.001         L4003                                */
/*     243        MIRCCT.001         L9001                                */
/*     250        MICNFG.001         C1205                                */
/*     252        MICNFG.001         C1305                                */
/*     500        MISL0AN.001        Dynamic External Input               */
/*     555        MISL0SY1.001       Dynamic External Input               */
/*     556        MISL0SY2.001       Dynamic External Input               */
/*     557        MISL0SY3.001       Dynamic External Input               */
/*     599        MICNFG.001         C1415                                */
/*     1101       MICNFG.001         C1005                                */
/*     1120       ActSched.001       Dynamic External Input               */
/*     1301       MIANCSSC.001       C0002                                */
/*     1304       MIANCAGP.001       L0002 - Path Dependent Static File   */
/*     1305       MIANPPAN.001       L1001                                */
/*     1306       MIRFOIAN.001       L1001                                */
/*     1334       MIB2GEOP.001       Dynamic Internal Input               */
/*     1500       MIANCARP.001       C0010                                */
/*     1501       MIANCARP.001       C0011                                */
/*     1502       MIANCARP.002       Dynamic External Input               */
/*     1503       MIANCARP.001       C0012                                */
/*     1984       MICNFG.001         C1105                                */
/*     10501      AM1EPHN0.001       Dynamic Internal Input               */
/*     10502      AM1ATTNF.001       Dynamic Internal Input               */
/*                                                                        */
/************************************************************************/


/*       PCF Entry for 190:MISANCGM                                       */
/*       MISR Ancillary Dataset for Camera Model  OBJECT = PCF_ENTRY
   CLASS = 11
   LOGICAL_ID = 190
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MISANCGM"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   CLOSEST_QUERY_OFFSET = "WEEKS=9"
   CLOSEST_QUERY_DIRECTION = "Backward"
   CLOSEST_QUERY_RETRIES = 6
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY
```

```
/*      PCF Entry for 227:MIRCCT                                         */
/*      MISR RC Thresholds datasetOBJECT = PCF_ENTRY
   CLASS = 12
   LOGICAL_ID = 227
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MIRCCT"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "L4003"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 243:MIRCCT                                         */
/*      MISR RC Thresholds datasetOBJECT = PCF_ENTRY
   CLASS = 13
   LOGICAL_ID = 243
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MIRCCT"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "L9001"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 500:MISL0AN                                        */
/*      L0 AN data                                                       */
OBJECT = PCF_ENTRY
   CLASS = 14
   LOGICAL_ID = 500
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MISL0AN"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 2
   INPUT_TYPE = "Required"
  /*  ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y" */
   ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
```

```
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
/* 4PY version */
      OBJECT = FILETYPE
         FILETYPE_NAME = "Multi-File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
      OBJECT = AUXILIARY_LOGICAL_ID
         CLASS = 1
         AUX_LOGICAL_ID = 501
      END_OBJECT = AUXILIARY_LOGICAL_ID
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 1120:ActSched                                           */
/*      Detailed Activity Schedule from EMOS                                  */
OBJECT = PCF_ENTRY
      CLASS = 16
      LOGICAL_ID = 1120
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "ActSched"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 2
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 1301:MIANCSSC                                           */
/*      MISR CSSC dataset                                                     */
OBJECT = PCF_ENTRY
      CLASS = 17
      LOGICAL_ID = 1301
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MIANCSSC"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      SCIENCE_GROUP = "C0002"
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY
```

```
/*      PCF Entry for 1304:MIANCAGP                                           */
/*      MISR Ancillary Geographic Product                                     */
OBJECT = PCF_ENTRY
   CLASS = 18
   LOGICAL_ID = 1304
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MIANCAGP"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "L0002"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Metadata"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
   OBJECT = METADATA_QUERY
      CLASS = 1
      PARM_NAME = "SP_AM_PATH_NO"
      OPERATOR = "=="
      VALUE = "999"
      DATABASE_QUERY = "PATH NUMBER"
   END_OBJECT = METADATA_QUERY
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 1305:MIANPPAN                                           */
/*      MISR Project Parameters (PP) dataset                                  */
OBJECT = PCF_ENTRY
   CLASS = 19
   LOGICAL_ID = 1305
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MIANPPAN"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "L1001"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Metadata"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
   OBJECT = METADATA_QUERY
      CLASS = 1
      PARM_NAME = "SP_AM_PATH_NO"
      OPERATOR = "=="
      VALUE = "999"
      DATABASE_QUERY = "PATH NUMBER"
   END_OBJECT = METADATA_QUERY
```

```
END_OBJECT = PCF_ENTRY



/*      PCF Entry for 555:MISL0SY1                                             */
/*      L0 Out of Synch data                                                   */
OBJECT = PCF_ENTRY
   CLASS = 20
   LOGICAL_ID = 555
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MISL0SY1"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 2
   INPUT_TYPE = "Optional"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Multi-File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
   OBJECT = AUXILIARY_LOGICAL_ID
      CLASS = 1
      AUX_LOGICAL_ID = 5551
   END_OBJECT = AUXILIARY_LOGICAL_ID
   OBJECT = OPTIONAL_INPUT
      CLASS = 1
      CATEGORY = "Out of Sync SY1"
      ORDER = 1
      RUNTIME_PARM_ID = 555
      TIMER = "SECS=10"
      TEMPORAL = "N"
   END_OBJECT = OPTIONAL_INPUT
END_OBJECT = PCF_ENTRY



/*      PCF Entry for 556:MISL0SY2                                             */
/*      L0 Out of Synch data                                                   */
OBJECT = PCF_ENTRY
   CLASS = 21
   LOGICAL_ID = 556
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MISL0SY2"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 2
   INPUT_TYPE = "Optional"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Multi-File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
   OBJECT = AUXILIARY_LOGICAL_ID
```

```
         CLASS = 1
         AUX_LOGICAL_ID = 5561
      END_OBJECT = AUXILIARY_LOGICAL_ID
      OBJECT = OPTIONAL_INPUT
         CLASS = 1
         CATEGORY = "Out of Sync SY2"
         ORDER = 1
         RUNTIME_PARM_ID = 556
         TIMER = "SECS=10"
         TEMPORAL = "N"
      END_OBJECT = OPTIONAL_INPUT
END_OBJECT = PCF_ENTRY




/*      PCF Entry for 557:MISL0SY3                                          */
/*      L0 Out of Synch data                                                */
OBJECT = PCF_ENTRY
   CLASS = 22
   LOGICAL_ID = 557
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MISL0SY3"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 2
   INPUT_TYPE = "Optional"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Multi-File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
   OBJECT = AUXILIARY_LOGICAL_ID
      CLASS = 1
      AUX_LOGICAL_ID = 5571
   END_OBJECT = AUXILIARY_LOGICAL_ID
   OBJECT = OPTIONAL_INPUT
      CLASS = 1
      CATEGORY = "Out of Sync SY3"
      ORDER = 1
      RUNTIME_PARM_ID = 557
      TIMER = "SECS=10"
      TEMPORAL = "N"
   END_OBJECT = OPTIONAL_INPUT
END_OBJECT = PCF_ENTRY




/*      PCF Entry for 1306:MIRFOIAN                                         */
/*      MISR Reference Orbit Imagery                                        */
OBJECT = PCF_ENTRY
   CLASS = 110
   LOGICAL_ID = 1306
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MIRFOIAN"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
```

```
      MAX_GRANULES_REQUIRED = 1
      SCIENCE_GROUP = "L1001"
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Metadata"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
      OBJECT = METADATA_QUERY
         CLASS = 1
         PARM_NAME = "SP_AM_PATH_NO"
         OPERATOR = "=="
         VALUE = "999"
         DATABASE_QUERY = "PATH NUMBER"
      END_OBJECT = METADATA_QUERY
END_OBJECT = PCF_ENTRY



/*      PCF Entry for 1334:MIB2GEOP                                     */
/*      MISR Geometric Parameters                                       */
OBJECT = PCF_ENTRY
   CLASS = 111
   LOGICAL_ID = 1334
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MIB2GEOP"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY



/*      PCF Entry for 1500:MIANCARP                                     */
/*      MISR Ancillary Radiometric Product                              */
OBJECT = PCF_ENTRY
   CLASS = 112
   LOGICAL_ID = 1500
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MIANCARP"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "C0010"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
```

```
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY



   /*      PCF Entry for 1501:MIANCARP                                        */
   /*      MISR Ancillary Radiometric Product                                */
   OBJECT = PCF_ENTRY
      CLASS = 113
      LOGICAL_ID = 1501
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MIANCARP"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      SCIENCE_GROUP = "C0011"
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY



   /*      PCF Entry for 1502:MIANCARP                                        */
   /*      MISR Ancillary Radiometric Product                                */
   OBJECT = PCF_ENTRY
      CLASS = 114
      LOGICAL_ID = 1502
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MIANCARP"
      DATA_TYPE_VERSION = "002"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      CLOSEST_QUERY_OFFSET = "WEEKS=8"
      CLOSEST_QUERY_DIRECTION = "Backward"
      CLOSEST_QUERY_RETRIES = 10
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
```

```
/*      PCF Entry for 1503:MIANCARP                                       */
/*      MISR Ancillary Radiometric Product                               */
OBJECT = PCF_ENTRY
   CLASS = 115
   LOGICAL_ID = 1503
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MIANCARP"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "C0012"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 250:MICNFG                                          */
/*      MISR RCCM configuration file                                      */
OBJECT = PCF_ENTRY
   CLASS = 116
   LOGICAL_ID = 250
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MICNFG"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "C1205"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 252:MICNFG                                          */
/*      MISR GRP configuration file                                       */
OBJECT = PCF_ENTRY
   CLASS = 117
   LOGICAL_ID = 252
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MICNFG"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
```

```
      SCIENCE_GROUP = "C1305"
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY




/*      PCF Entry for 599:MICNFG                                          */
/*      MISR RAP configuration file                                      */
OBJECT = PCF_ENTRY
   CLASS = 118
   LOGICAL_ID = 599
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MICNFG"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "C1415"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY




/*      PCF Entry for 1984:MICNFG                                         */
/*      MISR RP configuration file                                       */
OBJECT = PCF_ENTRY
   CLASS = 119
   LOGICAL_ID = 1984
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MICNFG"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "C1105"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY
```

```
/**** Attitude/Ephemeris/DEM entry.  Please delete if not used by PGE. **/
/*       PCF Entry for 10501:AM1EPHN0                                   */
/*       Ephemeris data generated from DPREP                            */
/*       External Data Source                                           */
OBJECT = PCF_ENTRY
   CLASS = 120
   LOGICAL_ID = 10501
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "AM1EPHN0"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 2
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY



/**** Attitude/Ephemeris/DEM entry.  Please delete if not used by PGE. **/
/*       PCF Entry for 10502:AM1ATTNF                                   */
/*       Attitude data generated by DPREP                               */
/*       External Data Source                                           */
OBJECT = PCF_ENTRY
   CLASS = 121
   LOGICAL_ID = 10502
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "AM1ATTNF"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 2
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY



/*       PCF Entry for 1101:MICNFG                                      */
/*       MISR PCS configuration file                                    */
OBJECT = PCF_ENTRY
   CLASS = 132
   LOGICAL_ID = 1101
   PCF_FILE_TYPE = 1
```

```
          DATA_TYPE_NAME = "MICNFG"
          DATA_TYPE_VERSION = "001"
          MIN_GRANULES_REQUIRED = 1
          MAX_GRANULES_REQUIRED = 1
          SCIENCE_GROUP = "C1005"
          INPUT_TYPE = "Required"
          NUMBER_NEEDED = 1
          QUERY_TYPE = "Temporal"
          SPATIAL_TIME_DELTA = 0
          KEY_INPUT = "N"
          OBJECT = FILETYPE
             FILETYPE_NAME = "Single File Granule"
             CLASS = 1
          END_OBJECT = FILETYPE
       END_OBJECT = PCF_ENTRY


/************************************************************************/
/*   MISR PGE 1 AN Outputs                                            */
/*      Output:                                                       */
/*      LID     ESDT.Version    Science Group    Associated MCF       */
/*      251     MIANRCCH.002    S4               1136                 */
/*      600     MIL1A.001       S5               1130                 */
/*      610     MISBR.002       S6               1138                 */
/*      611     MISBR.002       S7               1138                 */
/*      650     MISQA.002       S10              1137                 */
/*      1375    MI1B2T.001      S2               1133                 */
/*      1376    MI1B2E.001      S1               1134                 */
/*      1377    MIRCCM.001      S3               1135                 */
/*      1335    MISQA.002       S11              11371                */
/*      1336    MISQA.002       S12              11372                */
/*      1337    MISQA.002       S13              11373                */
/*      1976    MI1B1.001       S8               1140                 */
/*      1983    MIB1LM.001      S9               1131                 */
/*      1985    MISQA.002       S14              11374                */
/*      1986    MISQA.002       S15              11375                */
/*                                                                    */
/************************************************************************/


/*       PCF Entry for 251:MIANRCCH                                   */
/*       MISR RC Histogram file                                       */
OBJECT = PCF_ENTRY
   CLASS = 136
   LOGICAL_ID = 251
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MIANRCCH"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 1136
   SCIENCE_GROUP = "S4"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY
```

```
/*      PCF Entry for 600:MIL1A                                                */
/*      MISR L1A Product                                                       */
OBJECT = PCF_ENTRY
   CLASS = 137
   LOGICAL_ID = 600
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MIL1A"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 1130
   SCIENCE_GROUP = "S5"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 610:MISBR                                                */
/*      MISR Browse data HDF file                                             */
OBJECT = PCF_ENTRY
   CLASS = 138
   LOGICAL_ID = 610
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MISBR"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 1138
   SCIENCE_GROUP = "S6"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 611:MISBR                                                */
/*      MISR Browse data JPEG file                                            */
OBJECT = PCF_ENTRY
   CLASS = 139
   LOGICAL_ID = 611
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MISBR"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 1
```

```
      ASSOCIATED_MCF_ID = 1138
      SCIENCE_GROUP = "S7"
      INSTANCE = 0
      MINIMUM_SIZE = 0
      MAXIMUM_SIZE = 0
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY




/*      PCF Entry for 650:MISQA                                          */
/*      MISR L1A QA                                                      */
OBJECT = PCF_ENTRY
   CLASS = 140
   LOGICAL_ID = 650
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MISQA"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 1137
   SCIENCE_GROUP = "S10"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY




/*      PCF Entry for 1375:MI1B2T                                        */
/*      MISR L1B2 Terrain data                                           */
OBJECT = PCF_ENTRY
   CLASS = 141
   LOGICAL_ID = 1375
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MI1B2T"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 1133
   SCIENCE_GROUP = "S2"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   DISTINCT_VALUE = "AN"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY
```

```
/*      PCF Entry for 1376:MI1B2E                                            */
/*      MISR L1B2 Ellipsoid data                                            */
OBJECT = PCF_ENTRY
   CLASS = 142
   LOGICAL_ID = 1376
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MI1B2E"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 1134
   SCIENCE_GROUP = "S1"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   DISTINCT_VALUE = "AN"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 1377:MIRCCM                                           */
/*      MISR L1B2 RCCM data                                                 */
OBJECT = PCF_ENTRY
   CLASS = 143
   LOGICAL_ID = 1377
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MIRCCM"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 1135
   SCIENCE_GROUP = "S3"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   DISTINCT_VALUE = "AN"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 1335:MISQA                                            */
/*      MISR L1B2 Terrain QA data                                           */
OBJECT = PCF_ENTRY
   CLASS = 144
   LOGICAL_ID = 1335
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MISQA"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 1
```

```
      ASSOCIATED_MCF_ID = 11371
      SCIENCE_GROUP = "S11"
      INSTANCE = 0
      MINIMUM_SIZE = 0
      MAXIMUM_SIZE = 0
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 1336:MISQA                                          */
/*      MISR L1B2 Ellipsoid QA data                                       */
OBJECT = PCF_ENTRY
   CLASS = 145
   LOGICAL_ID = 1336
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MISQA"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 11372
   SCIENCE_GROUP = "S12"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 1337:MISQA                                          */
/*      MISR L1B2 RCCM QA data                                            */
OBJECT = PCF_ENTRY
   CLASS = 146
   LOGICAL_ID = 1337
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MISQA"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 11373
   SCIENCE_GROUP = "S13"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY


/*      PCF Entry for 1976:MI1B1                  `                       */
```

```
/*       MISR L1B1 Radiometric Product                              */
OBJECT = PCF_ENTRY
   CLASS = 147
   LOGICAL_ID = 1976
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MI1B1"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 1140
   SCIENCE_GROUP = "S8"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY



/*       PCF Entry for 1983:MIB1LM                                  */
/*       MISR L1B1 Local Mode data                                  */
OBJECT = PCF_ENTRY
   CLASS = 148
   LOGICAL_ID = 1983
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MIB1LM"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 6
   ASSOCIATED_MCF_ID = 1131
   SCIENCE_GROUP = "S9"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY



/*       PCF Entry for 1985:MISQA                                   */
/*       MISR L1B1 QA data                                          */
OBJECT = PCF_ENTRY
   CLASS = 149
   LOGICAL_ID = 1985
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MISQA"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 11374
   SCIENCE_GROUP = "S14"
   INSTANCE = 0
   MINIMUM_SIZE = 0
```

```
      MAXIMUM_SIZE = 0
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY



/*        PCF Entry for 1986:MISQA                                             */
/*        MISR L1B1 Local Mode QA                                             */
OBJECT = PCF_ENTRY
CLASS = 150
   LOGICAL_ID = 1986
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MISQA"
   DATA_TYPE_VERSION = "002"
   MIN_GRANULE_YIELD = 0
   MAX_GRANULE_YIELD = 6
   ASSOCIATED_MCF_ID = 11375
   SCIENCE_GROUP = "S15"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
     FILETYPE_NAME = "Single File Granule"
     CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY



OBJECT = PCF_ENTRY
   CLASS = 151
   LOGICAL_ID = 292
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Product version              "
   PGE_PARAMETER_DEFAULT = "0007"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

 OBJECT = PCF_ENTRY
   CLASS = 152
   LOGICAL_ID = 295
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Camera                       "
   PGE_PARAMETER_DEFAULT = "An"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY


OBJECT = PCF_ENTRY
   CLASS = 153
   LOGICAL_ID = 620
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Ascii met file for HDF browse"
   PGE_PARAMETER_DEFAULT = "610:1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
```

```
OBJECT = PCF_ENTRY
   CLASS = 154
   LOGICAL_ID = 621
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Ascii met file for JPEG browse"
   PGE_PARAMETER_DEFAULT = "611:1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY


OBJECT = PCF_ENTRY
   CLASS = 155
   LOGICAL_ID = 1102
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "which pge                    "
   PGE_PARAMETER_DEFAULT = "MISR_PGE01"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY


OBJECT = PCF_ENTRY
   CLASS = 156
   LOGICAL_ID = 1104
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Orbit number               "
   PGE_PARAMETER_DEFAULT = "999999"
   PGE_PARAMETER_DYNAMIC_VALUE = "ORBIT NUMBER"
END_OBJECT = PCF_ENTRY


OBJECT = PCF_ENTRY
   CLASS = 157
   LOGICAL_ID = 1103
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Orbit path                 "
   PGE_PARAMETER_DEFAULT = "999"
   PGE_PARAMETER_DYNAMIC_VALUE = "PATH NUMBER"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 158
   LOGICAL_ID = 10119
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Disabled status code list"
   PGE_PARAMETER_DEFAULT = "35870 163843126 163843127 163842611 163842612
166300169 164662287"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

END
```

A.3.2 MISR ESDT MISANCGM ODL

```
DATA_TYPE_NAME = "MISANCGM"
DATA_TYPE_VERSION = "002"
```

```
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "Camera Geometric Model for Level 1B2"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 1.0
PROCESSING_LEVEL = "L1"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "NONROUTINE"
SUPPLIER_NAME = "LARC"
/* BOUNDARY = "START_OF_YEAR" */
/* PERIOD = "YEARS=5"    */
DELAY = 1
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT = "LARC"
PROCESSED_AT = "LARC"

END
```

A.3.3 MISR ESDT MIRCCT ODL

```
DATA_TYPE_NAME = "MIRCCT"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Radiometric Camera-by-Camera Threshold dataset"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 10.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"

END
```

A.3.4 MISR ESDT MISL0AN ODL

```
DATA_TYPE_NAME = "MISL0AN"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Level 0 CCD Science Data AN Camera"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 1000.0
PROCESSING_LEVEL = "L0"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
```

```
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
/* PERIOD = "ORBITS=1" */
PERIOD = "HOURS=2"
/* BOUNDARY = "START_OF_ORBIT" */
BOUNDARY = "START_OF_DAY"
DURATION = "HOURS=2"
DELAY = 1
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"
OBJECT = FILETYPE
   CLASS = 2
   FILETYPE_NAME = "Multi-File Granule"
   MAXIMUM_NUM_FILES = 2
END_OBJECT = FILETYPE

END
```

A.3.5 MISR ESDT ActSched ODL

```
DATA_TYPE_NAME = "ActSched"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "Detailed Activity Schedule"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 1.0
/*  Changed by Jim Galasso 10/9/1999       */
/*  Processing Level cannot be L0 multiple files using the same LID     */
/*  Change of Processing level is to support PGE processing when 2 DAS  */
/*  files are required because the PGE's DPR times span 2 files         */
PROCESSING_LEVEL = "SCHED"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
/*  Q:  Should the supplier of the DAS be identified as EMOS?       */
SUPPLIER_NAME = "EMOS"
PERIOD = "DAYS=1"
/*  Boundary set for DAS files to be 2000 to 2000 each day       */
BOUNDARY = "START_OF_DAY-14400"
DURATION = "HOURS=24"
DELAY = 3600
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.6 MISR ESDT MIANCSSC ODL

```
DATA_TYPE_NAME = "MIANCSSC"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Cloud Screening Surface Classification dataset"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 5.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.7 MISR ESDT MIANCAGP ODL

```
DATA_TYPE_NAME = "MIANCAGP"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Ancillary Geographic Product"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 110.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "Y"
OBJECT = METADATA_DEFINITION
    CLASS = 1
    PARM_NAME = "SP_AM_PATH_NO"
    CONTAINER_NAME = "AdditionalAttributes"
    TYPE = "STR"
END_OBJECT = METADATA_DEFINITION
OBJECT = USE_OBJECT
      CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT

END
```

A.3.8 MISR ESDT MIANPPAN ODL

```
DATA_TYPE_NAME = "MIANPPAN"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Projection Parameters Ancillary Dataset, Camera
AN"
```

```
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 310.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "Y"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
OBJECT = METADATA_DEFINITION
    CLASS = 1
    PARM_NAME = "SP_AM_PATH_NO"
    CONTAINER_NAME = "AdditionalAttributes"
    TYPE = "STR"
END_OBJECT = METADATA_DEFINITION

END
```

A.3.9 MISR ESDT MISL0SY1 ODL

```
DATA_TYPE_NAME = "MISL0SY1"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Out of Sync L0 CCSDS packets for APID = 373"
PROVIDER = "Langley Research Center"
PROCESSING_LEVEL = "L0"
HDF_DATA = "N"
/*  Q: NOMINAL_SIZE ????   */
NOMINAL_SIZE = 5.9
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
PERIOD = "ORBITS=1"
/*  PERIOD = "HOURS=2" */
BOUNDARY = "START_OF_ORBIT"
/* BOUNDARY = "START_OF_DAY+3600" */
DURATION = "ORBITS=1"
/* DURATION = "HOURS=2"    */
DELAY = 1
SPATIAL_FLAG = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"
OBJECT = FILETYPE
    CLASS = 2
    FILETYPE_NAME = "Multi-File Granule"
    MAXIMUM_NUM_FILES = 2
END_OBJECT = FILETYPE

END
```

A.3.10 MISR ESDT MISL0SY2 ODL

```
DATA_TYPE_NAME = "MISL0SY2"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Out of Sync L0 CCSDS packets for APID = 374"
PROVIDER = "Langley Research Center"
PROCESSING_LEVEL = "L0"
HDF_DATA = "N"
/*  Q: NOMINAL_SIZE ????   */
NOMINAL_SIZE = 5.9
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
PERIOD = "ORBITS=1"
/* PERIOD = "HOURS=2" */
BOUNDARY = "START_OF_ORBIT"
/* BOUNDARY = "START_OF_DAY+3600" */
DURATION = "ORBITS=1"
/* DURATION = "HOURS=2"   */
DELAY = 1
SPATIAL_FLAG = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"
OBJECT = FILETYPE
   CLASS = 2
   FILETYPE_NAME = "Multi-File Granule"
   MAXIMUM_NUM_FILES = 2
END_OBJECT = FILETYPE

END
```

A.3.11 MISR ESDT MISL0SY3 ODL

```
DATA_TYPE_NAME = "MISL0SY3"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Out of Sync L0 CCSDS packets for APID = 378"
PROVIDER = "Langley Research Center"
PROCESSING_LEVEL = "L0"
HDF_DATA = "N"
/*  Q: NOMINAL_SIZE ????   */
NOMINAL_SIZE = 5.9
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
PERIOD = "ORBITS=1"
/* PERIOD = "HOURS=2" */
BOUNDARY = "START_OF_ORBIT"
/* BOUNDARY = "START_OF_DAY+3600" */
DURATION = "ORBITS=1"
/* DURATION = "HOURS=2"   */
```

```
DELAY = 1
SPATIAL_FLAG = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"
OBJECT = FILETYPE
    CLASS = 2
    FILETYPE_NAME = "Multi-File Granule"
    MAXIMUM_NUM_FILES = 2
END_OBJECT = FILETYPE

END
```

A.3.12   MISR ESDT MIRFOIAN ODL

```
DATA_TYPE_NAME = "MIRFOIAN"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Reference Orbit Imagery Ancillary Dataset,
Camera AN"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 280.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
OBJECT = METADATA_DEFINITION
    CLASS = 1
    PARM_NAME = "SP_AM_PATH_NO"
    CONTAINER_NAME = "AdditionalAttributes"
    TYPE = "STR"
END_OBJECT = METADATA_DEFINITION
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.13   MISR ESDT MIB2GEOP ODL

```
DATA_TYPE_NAME = "MIB2GEOP"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Geometric Parameters"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 6.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
```

```
DYNAMIC_FLAG = "I"
SPATIAL_FLAG  = "Y"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.14 MISR ESDT MIANCARP ODL (Version# 001)

```
DATA_TYPE_NAME = "MIANCARP"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Ancillary Radiometric Product (ARP)"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 5.0
PROCESSING_LEVEL = "All"
HDF_DATA = "Y"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT

END
```

A.3.15 MISR ESDT MIANCARP ODL (Version# 002)

```
DATA_TYPE_NAME = "MIANCARP"
DATA_TYPE_VERSION = "002"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Ancillary Radiometric Product (ARP)"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 5.0
PROCESSING_LEVEL = "All"
HDF_DATA = "Y"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "NONROUTINE"
/*  PERIOD = "MONTHS=2" */
/*  BOUNDARY = "START_OF_MONTH"  */
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT = "LARC"
PROCESSED_AT = "LARC"
```

END

A.3.16 MISR ESDT MICNFG ODL

```
DATA_TYPE_NAME = "MICNFG"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Configuration File for all PGEs"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 0.5
PROCESSING_LEVEL = "All"
HDF_DATA = "N"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
      CLASS = 1
      USED_BY = "LARC"
END_OBJECT = USE_OBJECT

END
```

A.3.17 MISR ESDT AM1EPHN0 ODL

```
DATA_TYPE_NAME = "AM1EPHN0"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "All"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "AM-1 L0/FDD Ephemeris data in Toolkit format"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 2.0
PROCESSING_LEVEL = "L1"
DYNAMIC_FLAG = "I"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
HDF_DATA = "N"

END
```

A.3.18 MISR ESDT AM1ATTNF ODL

```
DATA_TYPE_NAME = "AM1ATTNF"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "All"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "AM-1 FDD Attitude data in Toolkit format"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 2.0
PROCESSING_LEVEL = "L1"
```

```
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
HDF_DATA = "N"

END
```

A.3.19 MISR ESDT MIANRCCH ODL

```
DATA_TYPE_NAME = "MIANRCCH"
DATA_TYPE_VERSION = "002"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Radiometric Camera-by-Camera Histogram Dataset"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 3.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "N"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.20 MISR ESDT MIL1A ODL

```
DATA_TYPE_NAME = "MIL1A"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Level 1A CCD Science data, all cameras"
PROVIDER = "Langley Research Center"
/*    Q: Need to find the correct nominal file size for MIL1A          */
/* NOMINAL_SIZE = 498.0 */
/* NOMINAL_SIZE = 12000.0 */
/* NOMINAL_SIZE = 100.0 */
NOMINAL_SIZE = 1500.0
PROCESSING_LEVEL = "L1A"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "N"
```

```
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.21 MISR ESDT MISBR ODL

```
DATA_TYPE_NAME = "MISBR"
DATA_TYPE_VERSION = "002"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Browse data for use with systematic QA analysis"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 3.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.22 MISR ESDT MISQA ODL

```
DATA_TYPE_NAME = "MISQA"
DATA_TYPE_VERSION = "002"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Quality Assurance data"
PROVIDER = "Langley Research Center"
/*    Increased to 20.0 from 1.0 by Jim Galasso 10/9/1999         */
NOMINAL_SIZE = 20.0
/*    Changed to Processing Level all 10/9/1999             */
PROCESSING_LEVEL = "ALL"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
PERIOD = "ORBITS=1"
BOUNDARY = "START_OF_ORBIT"
DURATION = "HOURS=2"
DELAY = 1
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
```

```
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.23 MISR ESDT MI1B2T ODL

```
DATA_TYPE_NAME = "MI1B2T"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Level 1B2 Terrain Data"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 400.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "Y"
DISTINCT_PARAMETER  = "AssociatedSensorShortName"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"
OBJECT = METADATA_DEFINITION
   CLASS = 2
   PARM_NAME = "AssociatedSensorShortName"
   CONTAINER_NAME = "AssociatedPlatformInstrumentSensor"
   TYPE = "STR"
END_OBJECT = METADATA_DEFINITION

END
```

A.3.24 MISR ESDT MI1B2E ODL

```
DATA_TYPE_NAME = "MI1B2E"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Level 1B2 Ellipsoid Data"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 700.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "Y"
```

```
DISTINCT_PARAMETER   = "AssociatedSensorShortName"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"
OBJECT = METADATA_DEFINITION
    CLASS = 2
    PARM_NAME = "AssociatedSensorShortName"
    CONTAINER_NAME = "AssociatedPlatformInstrumentSensor"
    TYPE = "STR"
END_OBJECT = METADATA_DEFINITION

END
```

A.3.25 MISR ESDT MIRCCM ODL

```
DATA_TYPE_NAME = "MIRCCM"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR radiometric camera-by-camera Cloud Mask"
PROVIDER = "Langley Research Center"
NOMINAL_SIZE = 3.0
PROCESSING_LEVEL = "L1B2"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "Y"
DISTINCT_PARAMETER   = "AssociatedSensorShortName"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"
OBJECT = METADATA_DEFINITION
    CLASS = 2
    PARM_NAME = "AssociatedSensorShortName"
    CONTAINER_NAME = "AssociatedPlatformInstrumentSensor"
    TYPE = "STR"
END_OBJECT = METADATA_DEFINITION

END
```

A.3.26 MISR ESDT MI1B1 ODL

```
DATA_TYPE_NAME = "MI1B1"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Level 1B2 Ellipsoid Data"
PROVIDER = "Langley Research Center"
/* NOMINAL_SIZE = 574.0 */
/* changed for FILEWATCHER!    */
```

```
/* NOMINAL_SIZE = 12000.0 */
/* NOMINAL_SIZE = 100.0 */
NOMINAL_SIZE = 1500.0
PROCESSING_LEVEL = "L1B1"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.27   MISR ESDT MIB1LM ODL

```
DATA_TYPE_NAME = "MIB1LM"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MISR"
PLATFORM = "AM1"
DATA_TYPE_DESCRIPTION = "MISR Level 1B1 Local Mode Radiance Data"
PROVIDER = "Langley Reseach Center"
NOMINAL_SIZE = 100.0
PROCESSING_LEVEL = "L1"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "LARC"
DELAY = 1
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "LARC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "LARC"
PROCESSED_AT  = "LARC"

END
```

A.3.28   MISR ORBIT ODL

```
PLATFORM = "AM1"
OBJECT = ORBIT_MODEL
    CLASS = 1

/*   ORBIT_NUMBER = 7472            */
    ORBIT_NUMBER = 7496

/*  Cross correlate using PATHMAP_MISR.odl file */

/*   ORBIT_PATH_NUMBER = 28    */
```

```
      ORBIT_PATH_NUMBER = 52

/*    ORBIT_PERIOD = "SECS=5934"        */
      ORBIT_PERIOD = "SECS=5933"


/*    ORBIT_START = "05/14/2001 10:41:32"  */
      ORBIT_START = "05/16/2001 02:14:44"

END_OBJECT = ORBIT_MODEL

END
```

A.3.29 MISR PATHMAP ODL

```
PLATFORM = "AM1"
PATHMAP_NAME = "MISR"
OBJECT = PATHMAP_ENTRY
     CLASS = 1
     ABSOLUTE_PATH = 1
     MAPPED_PATH = 1
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 2
     ABSOLUTE_PATH = 2
     MAPPED_PATH = 17
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 3
     ABSOLUTE_PATH = 3
     MAPPED_PATH = 33
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 4
     ABSOLUTE_PATH = 4
     MAPPED_PATH = 49
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 5
     ABSOLUTE_PATH = 5
     MAPPED_PATH = 65
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 6
     ABSOLUTE_PATH = 6
     MAPPED_PATH = 81
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 7
     ABSOLUTE_PATH = 7
     MAPPED_PATH = 97
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 8
     ABSOLUTE_PATH = 8
     MAPPED_PATH = 113
END_OBJECT = PATHMAP_ENTRY
```

```
OBJECT = PATHMAP_ENTRY
    CLASS = 9
    ABSOLUTE_PATH = 9
    MAPPED_PATH = 129
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 10
    ABSOLUTE_PATH = 10
    MAPPED_PATH = 145
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 11
    ABSOLUTE_PATH = 11
    MAPPED_PATH = 161
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 12
    ABSOLUTE_PATH = 12
    MAPPED_PATH = 177
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 13
    ABSOLUTE_PATH = 13
    MAPPED_PATH = 193
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 14
    ABSOLUTE_PATH = 14
    MAPPED_PATH = 209
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 15
    ABSOLUTE_PATH = 15
    MAPPED_PATH = 225
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 16
    ABSOLUTE_PATH = 16
    MAPPED_PATH = 8
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 17
    ABSOLUTE_PATH = 17
    MAPPED_PATH = 24
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 18
    ABSOLUTE_PATH = 18
    MAPPED_PATH = 40
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 19
    ABSOLUTE_PATH = 19
    MAPPED_PATH = 56
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 20
    ABSOLUTE_PATH = 20
    MAPPED_PATH = 72
```

```
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 21
     ABSOLUTE_PATH = 21
     MAPPED_PATH = 88
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 22
     ABSOLUTE_PATH = 22
     MAPPED_PATH = 104
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 23
     ABSOLUTE_PATH = 23
     MAPPED_PATH = 120
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 24
     ABSOLUTE_PATH = 24
     MAPPED_PATH = 136
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 25
     ABSOLUTE_PATH = 25
     MAPPED_PATH = 152
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 26
     ABSOLUTE_PATH = 26
     MAPPED_PATH = 168
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 27
     ABSOLUTE_PATH = 27
     MAPPED_PATH = 184
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 28
     ABSOLUTE_PATH = 28
     MAPPED_PATH = 200
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 29
     ABSOLUTE_PATH = 29
     MAPPED_PATH = 216
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 30
     ABSOLUTE_PATH = 30
     MAPPED_PATH = 232
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 31
     ABSOLUTE_PATH = 31
     MAPPED_PATH = 15
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 32
     ABSOLUTE_PATH = 32
```

```
          MAPPED_PATH = 31
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 33
     ABSOLUTE_PATH = 33
     MAPPED_PATH = 47
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 34
     ABSOLUTE_PATH = 34
     MAPPED_PATH = 63
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 35
     ABSOLUTE_PATH = 35
     MAPPED_PATH = 79
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 36
     ABSOLUTE_PATH = 36
     MAPPED_PATH = 95
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 37
     ABSOLUTE_PATH = 37
     MAPPED_PATH = 111
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 38
     ABSOLUTE_PATH = 38
     MAPPED_PATH = 127
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 39
     ABSOLUTE_PATH = 39
     MAPPED_PATH = 143
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 40
     ABSOLUTE_PATH = 40
     MAPPED_PATH = 159
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 41
     ABSOLUTE_PATH = 41
     MAPPED_PATH = 175
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 42
     ABSOLUTE_PATH = 42
     MAPPED_PATH = 191
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 43
     ABSOLUTE_PATH = 43
     MAPPED_PATH = 207
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 44
```

```
                ABSOLUTE_PATH = 44
                MAPPED_PATH = 223
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 45
                ABSOLUTE_PATH = 45
                MAPPED_PATH = 6
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 46
                ABSOLUTE_PATH = 46
                MAPPED_PATH = 22
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 47
                ABSOLUTE_PATH = 47
                MAPPED_PATH = 38
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 48
                ABSOLUTE_PATH = 48
                MAPPED_PATH = 54
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 49
                ABSOLUTE_PATH = 49
                MAPPED_PATH = 70
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 50
                ABSOLUTE_PATH = 50
                MAPPED_PATH = 86
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 51
                ABSOLUTE_PATH = 51
                MAPPED_PATH = 102
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 52
                ABSOLUTE_PATH = 52
                MAPPED_PATH = 118
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 53
                ABSOLUTE_PATH = 53
                MAPPED_PATH = 134
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 54
                ABSOLUTE_PATH = 54
                MAPPED_PATH = 150
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
                CLASS = 55
                ABSOLUTE_PATH = 55
                MAPPED_PATH = 166
        END_OBJECT = PATHMAP_ENTRY
        OBJECT = PATHMAP_ENTRY
```

```
        CLASS = 56
        ABSOLUTE_PATH = 56
        MAPPED_PATH = 182
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 57
        ABSOLUTE_PATH = 57
        MAPPED_PATH = 198
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 58
        ABSOLUTE_PATH = 58
        MAPPED_PATH = 214
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 59
        ABSOLUTE_PATH = 59
        MAPPED_PATH = 230
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 60
        ABSOLUTE_PATH = 60
        MAPPED_PATH = 13
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 61
        ABSOLUTE_PATH = 61
        MAPPED_PATH = 29
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 62
        ABSOLUTE_PATH = 62
        MAPPED_PATH = 45
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 63
        ABSOLUTE_PATH = 63
        MAPPED_PATH = 61
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 64
        ABSOLUTE_PATH = 64
        MAPPED_PATH = 77
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 65
        ABSOLUTE_PATH = 65
        MAPPED_PATH = 93
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 66
        ABSOLUTE_PATH = 66
        MAPPED_PATH = 109
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 67
        ABSOLUTE_PATH = 67
        MAPPED_PATH = 125
END_OBJECT = PATHMAP_ENTRY
```

```
OBJECT = PATHMAP_ENTRY
    CLASS = 68
    ABSOLUTE_PATH = 68
    MAPPED_PATH = 141
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 69
    ABSOLUTE_PATH = 69
    MAPPED_PATH = 157
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 70
    ABSOLUTE_PATH = 70
    MAPPED_PATH = 173
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 71
    ABSOLUTE_PATH = 71
    MAPPED_PATH = 189
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 72
    ABSOLUTE_PATH = 72
    MAPPED_PATH = 205
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 73
    ABSOLUTE_PATH = 73
    MAPPED_PATH = 221
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 74
    ABSOLUTE_PATH = 74
    MAPPED_PATH = 4
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 75
    ABSOLUTE_PATH = 75
    MAPPED_PATH = 20
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 76
    ABSOLUTE_PATH = 76
    MAPPED_PATH = 36
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 77
    ABSOLUTE_PATH = 77
    MAPPED_PATH = 52
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 78
    ABSOLUTE_PATH = 78
    MAPPED_PATH = 68
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 79
    ABSOLUTE_PATH = 79
    MAPPED_PATH = 84
```

```
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 80
    ABSOLUTE_PATH = 80
    MAPPED_PATH = 100
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 81
    ABSOLUTE_PATH = 81
    MAPPED_PATH = 116
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 82
    ABSOLUTE_PATH = 82
    MAPPED_PATH = 132
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 83
    ABSOLUTE_PATH = 83
    MAPPED_PATH = 148
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 84
    ABSOLUTE_PATH = 84
    MAPPED_PATH = 164
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 85
    ABSOLUTE_PATH = 85
    MAPPED_PATH = 180
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 86
    ABSOLUTE_PATH = 86
    MAPPED_PATH = 196
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 87
    ABSOLUTE_PATH = 87
    MAPPED_PATH = 212
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 88
    ABSOLUTE_PATH = 88
    MAPPED_PATH = 228
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 89
    ABSOLUTE_PATH = 89
    MAPPED_PATH = 11
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 90
    ABSOLUTE_PATH = 90
    MAPPED_PATH = 27
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 91
    ABSOLUTE_PATH = 91
```

```
        MAPPED_PATH = 43
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 92
        ABSOLUTE_PATH = 92
        MAPPED_PATH = 59
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 93
        ABSOLUTE_PATH = 93
        MAPPED_PATH = 75
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 94
        ABSOLUTE_PATH = 94
        MAPPED_PATH = 91
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 95
        ABSOLUTE_PATH = 95
        MAPPED_PATH = 107
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 96
        ABSOLUTE_PATH = 96
        MAPPED_PATH = 123
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 97
        ABSOLUTE_PATH = 97
        MAPPED_PATH = 139
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 98
        ABSOLUTE_PATH = 98
        MAPPED_PATH = 155
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 99
        ABSOLUTE_PATH = 99
        MAPPED_PATH = 171
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 100
        ABSOLUTE_PATH = 100
        MAPPED_PATH = 187
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 101
        ABSOLUTE_PATH = 101
        MAPPED_PATH = 203
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 102
        ABSOLUTE_PATH = 102
        MAPPED_PATH = 219
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 103
```

```
        ABSOLUTE_PATH = 103
        MAPPED_PATH = 2
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 104
        ABSOLUTE_PATH = 104
        MAPPED_PATH = 18
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 105
        ABSOLUTE_PATH = 105
        MAPPED_PATH = 34
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 106
        ABSOLUTE_PATH = 106
        MAPPED_PATH = 50
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 107
        ABSOLUTE_PATH = 107
        MAPPED_PATH = 66
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 108
        ABSOLUTE_PATH = 108
        MAPPED_PATH = 82
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 109
        ABSOLUTE_PATH = 109
        MAPPED_PATH = 98
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 110
        ABSOLUTE_PATH = 110
        MAPPED_PATH = 114
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 111
        ABSOLUTE_PATH = 111
        MAPPED_PATH = 130
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 112
        ABSOLUTE_PATH = 112
        MAPPED_PATH = 146
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 113
        ABSOLUTE_PATH = 113
        MAPPED_PATH = 162
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 114
        ABSOLUTE_PATH = 114
        MAPPED_PATH = 178
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
```

```
         CLASS = 115
         ABSOLUTE_PATH = 115
         MAPPED_PATH = 194
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 116
         ABSOLUTE_PATH = 116
         MAPPED_PATH = 210
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 117
         ABSOLUTE_PATH = 117
         MAPPED_PATH = 226
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 118
         ABSOLUTE_PATH = 118
         MAPPED_PATH = 9
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 119
         ABSOLUTE_PATH = 119
         MAPPED_PATH = 25
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 120
         ABSOLUTE_PATH = 120
         MAPPED_PATH = 41
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 121
         ABSOLUTE_PATH = 121
         MAPPED_PATH = 57
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 122
         ABSOLUTE_PATH = 122
         MAPPED_PATH = 73
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 123
         ABSOLUTE_PATH = 123
         MAPPED_PATH = 89
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 124
         ABSOLUTE_PATH = 124
         MAPPED_PATH = 105
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 125
         ABSOLUTE_PATH = 125
         MAPPED_PATH = 121
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
         CLASS = 126
         ABSOLUTE_PATH = 126
         MAPPED_PATH = 137
END_OBJECT = PATHMAP_ENTRY
```

```
OBJECT = PATHMAP_ENTRY
    CLASS = 127
    ABSOLUTE_PATH = 127
    MAPPED_PATH = 153
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 128
    ABSOLUTE_PATH = 128
    MAPPED_PATH = 169
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 129
    ABSOLUTE_PATH = 129
    MAPPED_PATH = 185
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 130
    ABSOLUTE_PATH = 130
    MAPPED_PATH = 201
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 131
    ABSOLUTE_PATH = 131
    MAPPED_PATH = 217
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 132
    ABSOLUTE_PATH = 132
    MAPPED_PATH = 233
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 133
    ABSOLUTE_PATH = 133
    MAPPED_PATH = 16
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 134
    ABSOLUTE_PATH = 134
    MAPPED_PATH = 32
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 135
    ABSOLUTE_PATH = 135
    MAPPED_PATH = 48
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 136
    ABSOLUTE_PATH = 136
    MAPPED_PATH = 64
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 137
    ABSOLUTE_PATH = 137
    MAPPED_PATH = 80
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 138
    ABSOLUTE_PATH = 138
    MAPPED_PATH = 96
```

```
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 139
     ABSOLUTE_PATH = 139
     MAPPED_PATH = 112
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 140
     ABSOLUTE_PATH = 140
     MAPPED_PATH = 128
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 141
     ABSOLUTE_PATH = 141
     MAPPED_PATH = 144
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 142
     ABSOLUTE_PATH = 142
     MAPPED_PATH = 160
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 143
     ABSOLUTE_PATH = 143
     MAPPED_PATH = 176
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 144
     ABSOLUTE_PATH = 144
     MAPPED_PATH = 192
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 145
     ABSOLUTE_PATH = 145
     MAPPED_PATH = 208
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 146
     ABSOLUTE_PATH = 146
     MAPPED_PATH = 224
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 147
     ABSOLUTE_PATH = 147
     MAPPED_PATH = 7
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 148
     ABSOLUTE_PATH = 148
     MAPPED_PATH = 23
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 149
     ABSOLUTE_PATH = 149
     MAPPED_PATH = 39
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 150
     ABSOLUTE_PATH = 150
```

```
     MAPPED_PATH = 55
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 151
     ABSOLUTE_PATH = 151
     MAPPED_PATH = 71
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 152
     ABSOLUTE_PATH = 152
     MAPPED_PATH = 87
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 153
     ABSOLUTE_PATH = 153
     MAPPED_PATH = 103
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 154
     ABSOLUTE_PATH = 154
     MAPPED_PATH = 119
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 155
     ABSOLUTE_PATH = 155
     MAPPED_PATH = 135
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 156
     ABSOLUTE_PATH = 156
     MAPPED_PATH = 151
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 157
     ABSOLUTE_PATH = 157
     MAPPED_PATH = 167
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 158
     ABSOLUTE_PATH = 158
     MAPPED_PATH = 183
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 159
     ABSOLUTE_PATH = 159
     MAPPED_PATH = 199
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 160
     ABSOLUTE_PATH = 160
     MAPPED_PATH = 215
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 161
     ABSOLUTE_PATH = 161
     MAPPED_PATH = 231
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 162
```

```
        ABSOLUTE_PATH = 162
        MAPPED_PATH = 14
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 163
        ABSOLUTE_PATH = 163
        MAPPED_PATH = 30
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 164
        ABSOLUTE_PATH = 164
        MAPPED_PATH = 46
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 165
        ABSOLUTE_PATH = 165
        MAPPED_PATH = 62
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 166
        ABSOLUTE_PATH = 166
        MAPPED_PATH = 78
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 167
        ABSOLUTE_PATH = 167
        MAPPED_PATH = 94
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 168
        ABSOLUTE_PATH = 168
        MAPPED_PATH = 110
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 169
        ABSOLUTE_PATH = 169
        MAPPED_PATH = 126
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 170
        ABSOLUTE_PATH = 170
        MAPPED_PATH = 142
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 171
        ABSOLUTE_PATH = 171
        MAPPED_PATH = 158
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 172
        ABSOLUTE_PATH = 172
        MAPPED_PATH = 174
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
        CLASS = 173
        ABSOLUTE_PATH = 173
        MAPPED_PATH = 190
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
```

```
      CLASS = 174
      ABSOLUTE_PATH = 174
      MAPPED_PATH = 206
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 175
      ABSOLUTE_PATH = 175
      MAPPED_PATH = 222
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 176
      ABSOLUTE_PATH = 176
      MAPPED_PATH = 5
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 177
      ABSOLUTE_PATH = 177
      MAPPED_PATH = 21
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 178
      ABSOLUTE_PATH = 178
      MAPPED_PATH = 37
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 179
      ABSOLUTE_PATH = 179
      MAPPED_PATH = 53
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 180
      ABSOLUTE_PATH = 180
      MAPPED_PATH = 69
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 181
      ABSOLUTE_PATH = 181
      MAPPED_PATH = 85
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 182
      ABSOLUTE_PATH = 182
      MAPPED_PATH = 101
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 183
      ABSOLUTE_PATH = 183
      MAPPED_PATH = 117
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 184
      ABSOLUTE_PATH = 184
      MAPPED_PATH = 133
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
      CLASS = 185
      ABSOLUTE_PATH = 185
      MAPPED_PATH = 149
END_OBJECT = PATHMAP_ENTRY
```

```
OBJECT = PATHMAP_ENTRY
    CLASS = 186
    ABSOLUTE_PATH = 186
    MAPPED_PATH = 165
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 187
    ABSOLUTE_PATH = 187
    MAPPED_PATH = 181
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 188
    ABSOLUTE_PATH = 188
    MAPPED_PATH = 197
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 189
    ABSOLUTE_PATH = 189
    MAPPED_PATH = 213
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 190
    ABSOLUTE_PATH = 190
    MAPPED_PATH = 229
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 191
    ABSOLUTE_PATH = 191
    MAPPED_PATH = 12
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 192
    ABSOLUTE_PATH = 192
    MAPPED_PATH = 28
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 193
    ABSOLUTE_PATH = 193
    MAPPED_PATH = 44
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 194
    ABSOLUTE_PATH = 194
    MAPPED_PATH = 60
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 195
    ABSOLUTE_PATH = 195
    MAPPED_PATH = 76
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 196
    ABSOLUTE_PATH = 196
    MAPPED_PATH = 92
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
    CLASS = 197
    ABSOLUTE_PATH = 197
    MAPPED_PATH = 108
```

```
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 198
     ABSOLUTE_PATH = 198
     MAPPED_PATH = 124
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 199
     ABSOLUTE_PATH = 199
     MAPPED_PATH = 140
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 200
     ABSOLUTE_PATH = 200
     MAPPED_PATH = 156
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 201
     ABSOLUTE_PATH = 201
     MAPPED_PATH = 172
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 202
     ABSOLUTE_PATH = 202
     MAPPED_PATH = 188
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 203
     ABSOLUTE_PATH = 203
     MAPPED_PATH = 204
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 204
     ABSOLUTE_PATH = 204
     MAPPED_PATH = 220
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 205
     ABSOLUTE_PATH = 205
     MAPPED_PATH = 3
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 206
     ABSOLUTE_PATH = 206
     MAPPED_PATH = 19
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 207
     ABSOLUTE_PATH = 207
     MAPPED_PATH = 35
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 208
     ABSOLUTE_PATH = 208
     MAPPED_PATH = 51
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 209
     ABSOLUTE_PATH = 209
```

```
     MAPPED_PATH = 67
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 210
     ABSOLUTE_PATH = 210
     MAPPED_PATH = 83
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 211
     ABSOLUTE_PATH = 211
     MAPPED_PATH = 99
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 212
     ABSOLUTE_PATH = 212
     MAPPED_PATH = 115
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 213
     ABSOLUTE_PATH = 213
     MAPPED_PATH = 131
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 214
     ABSOLUTE_PATH = 214
     MAPPED_PATH = 147
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 215
     ABSOLUTE_PATH = 215
     MAPPED_PATH = 163
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 216
     ABSOLUTE_PATH = 216
     MAPPED_PATH = 179
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 217
     ABSOLUTE_PATH = 217
     MAPPED_PATH = 195
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 218
     ABSOLUTE_PATH = 218
     MAPPED_PATH = 211
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 219
     ABSOLUTE_PATH = 219
     MAPPED_PATH = 227
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 220
     ABSOLUTE_PATH = 220
     MAPPED_PATH = 10
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 221
```

```
     ABSOLUTE_PATH = 221
     MAPPED_PATH = 26
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 222
     ABSOLUTE_PATH = 222
     MAPPED_PATH = 42
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 223
     ABSOLUTE_PATH = 223
     MAPPED_PATH = 58
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 224
     ABSOLUTE_PATH = 224
     MAPPED_PATH = 74
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 225
     ABSOLUTE_PATH = 225
     MAPPED_PATH = 90
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 226
     ABSOLUTE_PATH = 226
     MAPPED_PATH = 106
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 227
     ABSOLUTE_PATH = 227
     MAPPED_PATH = 122
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 228
     ABSOLUTE_PATH = 228
     MAPPED_PATH = 138
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 229
     ABSOLUTE_PATH = 229
     MAPPED_PATH = 154
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 230
     ABSOLUTE_PATH = 230
     MAPPED_PATH = 170
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 231
     ABSOLUTE_PATH = 231
     MAPPED_PATH = 186
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
     CLASS = 232
     ABSOLUTE_PATH = 232
     MAPPED_PATH = 202
END_OBJECT = PATHMAP_ENTRY
OBJECT = PATHMAP_ENTRY
```

```
    CLASS = 233
    ABSOLUTE_PATH = 233
    MAPPED_PATH = 218
END_OBJECT = PATHMAP_ENTRY
END
```

A.4   Typical Terra MODIS PGE & ESDT ODL Files

Listings are provided for the following MODIS ODL files:

A.4.1 MODIS PGE ODL for PGE_NAME PGE01

A.4.2 MODIS ESDT MOD000 ODL

A.4.3 MODIS ESDT MOD01 ODL

A.4.4 MODIS ESDT MOD01LUT ODL

A.4.5 MODIS ESDT MOD03 ODL

A.4.6 MODIS ESDT MOD03LUT ODL

A.4.7 MODIS PGE ODL for PGE_NAME PGE03

A.4.8 GDAS_0ZF ODL

A.4.9 OZ_DAILY ODL

A.4.10 REYNSST ODL

A.4.11 SEA_ICE ODL

A.4.12 NISE ODL

```
A typical MODIS PGE will differ from the examples here by the PGE_NAME, the
specific input/output files referenced, and runtime parameters.  However, the
overall structure of a given MODIS PGE ODL file would be similar to the ones
used here. (N.B. The ODL files shown here are associated with the MODIS
version 2.1 software)
```

A.4.1 MODIS PGE PGE01 ODL

```
PGE_NAME = "PGE01"
PGE_VERSION = "2.1"
PROFILE_ID = 1
PROFILE_DESCRIPTION = "The profile for MOD_PR01 and MOD_PR03 "
PLATFORM = "AM1"
INSTRUMENT = "MODIS"
MINIMUM_OUTPUTS = 0
SCHEDULE_TYPE = "Time"
PROCESSING_PERIOD = "MINS=15"
PROCESSING_BOUNDARY = "START_OF_MIN"
PGE_SSW_VERSION = "2.1"
QUERY_DELAY = 0

OBJECT = EXIT_MESSAGE
   CLASS= 1
   EXIT_CODE = 0
```

```
         EXIT_MESSAGE = "PGE01 Exit"
      END_OBJECT = EXIT_MESSAGE
      OBJECT = EXIT_DEPENDENCY
         CLASS= 1
         DEPENDENCY_PGE_NAME = "none"
         DEPENDENCY_SSW_VERSION = "none"
         EXIT_OPERATION = "="
         EXIT_CODE = 0
      END_OBJECT = EXIT_DEPENDENCY

      OBJECT = PCF_ENTRY
         CLASS = 10
         LOGICAL_ID = 599001
         PCF_FILE_TYPE = 1
         DATA_TYPE_NAME = "MOD000"
         DATA_TYPE_VERSION = "001"
         MIN_GRANULES_REQUIRED = 1
         MAX_GRANULES_REQUIRED = 1
         BEGIN_PERIOD_OFFSET = -7200
         END_PERIOD_OFFSET = -7200
         INPUT_TYPE = "Optional"
         ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
         NUMBER_NEEDED = 1
         QUERY_TYPE = "Temporal"
         SPATIAL_TIME_DELTA = 0
         KEY_INPUT = "N"
         OBJECT = FILETYPE
            FILETYPE_NAME = "L0 Data Files"
            CLASS = 1
         END_OBJECT = FILETYPE
         OBJECT = OPTIONAL_INPUT
            CLASS = 1
            ORDER = 1
            RUNTIME_PARM_ID = 51
            TIMER = "HOURS=4"
            TEMPORAL = "N"
         END_OBJECT = OPTIONAL_INPUT
      END_OBJECT = PCF_ENTRY

      OBJECT = PCF_ENTRY
         CLASS = 11
         LOGICAL_ID = 599002
         PCF_FILE_TYPE = 1
         DATA_TYPE_NAME = "MOD000"
         DATA_TYPE_VERSION = "001"
         MIN_GRANULES_REQUIRED = 1
         MAX_GRANULES_REQUIRED = 1
         BEGIN_PERIOD_OFFSET = 0
         END_PERIOD_OFFSET = 0
         INPUT_TYPE = "Required"
         ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
         NUMBER_NEEDED = 1
         QUERY_TYPE = "Temporal"
         SPATIAL_TIME_DELTA = 0
         KEY_INPUT = "N"
         OBJECT = FILETYPE
            FILETYPE_NAME = "L0 Data Files"
            CLASS = 1
         END_OBJECT = FILETYPE
```

```
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 12
   LOGICAL_ID = 599003
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MOD01LUT"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   BEGIN_PERIOD_OFFSET = 0
   END_PERIOD_OFFSET = 0
   SCIENCE_GROUP = "L1"
   INPUT_TYPE = "Required"
   ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 15
   LOGICAL_ID = 600020
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "MOD01LUT"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   BEGIN_PERIOD_OFFSET = 0
   END_PERIOD_OFFSET = 0
   SCIENCE_GROUP = "L2"
   INPUT_TYPE = "Required"
   ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   SPATIAL_TIME_DELTA = 0
   KEY_INPUT = "N"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Two GEO_parameter data files"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 17
   LOGICAL_ID = 10501
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "AM1EPHN0"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   BEGIN_PERIOD_OFFSET = 0
   END_PERIOD_OFFSET = 0
   INPUT_TYPE = "Required"
```

```
        ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
        NUMBER_NEEDED = 1
        QUERY_TYPE = "Temporal"
        SPATIAL_TIME_DELTA = 0
        KEY_INPUT = "N"
        OBJECT = FILETYPE
            FILETYPE_NAME = "Single File Granule"
            CLASS = 1
        END_OBJECT = FILETYPE
    END_OBJECT = PCF_ENTRY

    OBJECT = PCF_ENTRY
        CLASS = 18
        LOGICAL_ID = 10502
        PCF_FILE_TYPE = 1
        DATA_TYPE_NAME = "AM1ATTN0"
        DATA_TYPE_VERSION = "001"
        MIN_GRANULES_REQUIRED = 1
        MAX_GRANULES_REQUIRED = 1
        BEGIN_PERIOD_OFFSET = 0
        END_PERIOD_OFFSET = 0
        INPUT_TYPE = "Required"
        ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
        NUMBER_NEEDED = 1
        QUERY_TYPE = "Temporal"
        SPATIAL_TIME_DELTA = 0
        KEY_INPUT = "N"
        OBJECT = FILETYPE
            FILETYPE_NAME = "Single File Granule"
            CLASS = 1
        END_OBJECT = FILETYPE
    END_OBJECT = PCF_ENTRY

    OBJECT = PCF_ENTRY
        CLASS = 110
        LOGICAL_ID = 500100
        PCF_FILE_TYPE = 2
        DATA_TYPE_NAME = "MOD01"
        DATA_TYPE_VERSION = "001"
        MIN_GRANULE_YIELD = 3
        MAX_GRANULE_YIELD = 3
        ASSOCIATED_MCF_ID = 500500
        SCIENCE_GROUP = "S1"
        INSTANCE = 0
        MINIMUM_SIZE = 0
        MAXIMUM_SIZE = 0
        OBJECT = FILETYPE
            FILETYPE_NAME = "Single File Granule"
            CLASS = 1
        END_OBJECT = FILETYPE
    END_OBJECT = PCF_ENTRY

    OBJECT = PCF_ENTRY
        CLASS = 111
        LOGICAL_ID = 600000
        PCF_FILE_TYPE = 2
        DATA_TYPE_NAME = "MOD03"
        DATA_TYPE_VERSION = "001"
        MIN_GRANULE_YIELD = 3
```

```
      MAX_GRANULE_YIELD = 3
      ASSOCIATED_MCF_ID = 600111
      SCIENCE_GROUP = "S2"
      INSTANCE = 0
      MINIMUM_SIZE = 0
      MAXIMUM_SIZE = 0
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 113
   LOGICAL_ID = 503000
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Length of L1A granules in seconds"
   PGE_PARAMETER_DEFAULT = "300.000000"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 114
   LOGICAL_ID = 504000
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Scan rate for L1A granule"
   PGE_PARAMETER_DEFAULT = "1.477"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 115
   LOGICAL_ID = 505000
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "PGE version for L1A granule"
   PGE_PARAMETER_DEFAULT = "2.1.1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 116
   LOGICAL_ID = 800510
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "SatelliteInstrument; AM1M-Terra, PM1M-Aqua"
   PGE_PARAMETER_DEFAULT = "AM1M"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 117
   LOGICAL_ID = 800500
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "PGE01 Version"
   PGE_PARAMETER_DEFAULT = "2.1.1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 118
```

```
      LOGICAL_ID = 600280
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Source for spacecraft kinematic state"
      PGE_PARAMETER_DEFAULT = "SDP Toolkit"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
      CLASS = 119
      LOGICAL_ID = 600310
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Terrain Correction Flag"
      PGE_PARAMETER_DEFAULT = "TRUE"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
      CLASS = 120
      LOGICAL_ID = 600001
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "LOCALVERSIONID"
      PGE_PARAMETER_DEFAULT = "2.1.2"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY

END
```

A.4.2 MODIS ESDT MOD000 ODL

```
DATA_TYPE_NAME = "MOD000"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MODIS"
PLATFORM = "EOSAM1"
DATA_TYPE_DESCRIPTION = "L0 Input of PGE MOD_PR01"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 569.0
PROCESSING_LEVEL = "L0"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "GSFC"
PERIOD = "HOURS=2"
BOUNDARY = "START_OF_DAY"
DELAY = 43200
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
      CLASS = 1
      USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
OBJECT = FILETYPE
      CLASS = 1
      FILETYPE_NAME = "L0 Data Files"
      MAXIMUM_NUM_FILES = 6
END_OBJECT = USE_OBJECT
ARCHIVED_AT = "GSFC"
PROCESSED_AT = "GSFC"

END
```

A.4.3 MODIS ESDT MOD01 ODL

```
DATA_TYPE_NAME = "MOD01"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MODIS"
PLATFORM = "EOSAM1"
DATA_TYPE_DESCRIPTION = "An Input of PGE MOD_PR02"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 569.0
PROCESSING_LEVEL = "L1A"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "GSFC"
PERIOD = "MINS=5"
BOUNDARY = "START_OF_MIN"
DELAY = 43200
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
END
```

A.4.4 MODIS ESDT MOD01LUT ODL

```
DATA_TYPE_NAME = "MOD01LUT"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MODIS"
PLATFORM = "EOSAM1"
DATA_TYPE_DESCRIPTION = "An Input (static) of PGE MOD01"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 0.357
PROCESSING_LEVEL = "L1"
HDF_DATA = "N"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
END
```

A.4.5 MODIS ESDT MOD03 ODL

```
DATA_TYPE_NAME = "MOD03"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MODIS"
PLATFORM = "EOSAM1"
DATA_TYPE_DESCRIPTION = "Input/Output of PGE MOD_PR29/MOD_PR03"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 58.0
PROCESSING_LEVEL = "Geo"
HDF_DATA = "Y"
DYNAMIC_FLAG = "I"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "GSFC"
```

```
PERIOD = "MINS=5"
BOUNDARY = "START_OF_MIN"
DELAY = 43200
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
END
```

A.4.6 MODIS ESDT MOD03LUT ODL

```
DATA_TYPE_NAME = "MOD03LUT"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MODIS"
PLATFORM = "EOSAM1"
DATA_TYPE_DESCRIPTION = "An Input (static) of PGE MOD_PR03"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 0.357
PROCESSING_LEVEL = "L1"
HDF_DATA = "N"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
OBJECT = FILETYPE
    CLASS = 1
    FILETYPE_NAME = "Two GEO_parameter data files"
    MAXIMUM_NUM_FILES = 2
END_OBJECT = FILETYPE
END
```

A.4.7 MODIS PGE PGE03 ODL

```
PGE_NAME = "TerraPGE03"
PGE_VERSION = "3.0.0"
PROFILE_ID = 1
PROFILE_DESCRIPTION = "First Step in Level 2 Processing"
PGE_DEFAULT_PROFILE = "N"
PLATFORM = "AM1"
INSTRUMENT = "MODIS"
MINIMUM_OUTPUTS = 0
SCHEDULE_TYPE = "Time"
PROCESSING_PERIOD = "MINS=5"
PROCESSING_BOUNDARY = "START_OF_MIN"
PGE_SSW_VERSION = "3.0.0"
QUERY_DELAY = 0
OBJECT = EXIT_MESSAGE
    CLASS= 1
    EXIT_CODE = 0
    EXIT_MESSAGE = "none"
END_OBJECT = EXIT_MESSAGE
OBJECT = EXIT_DEPENDENCY
    CLASS= 1
    DEPENDENCY_PGE_NAME = "none"
```

```
      DEPENDENCY_SSW_VERSION = "none"
      EXIT_OPERATION = "="
      EXIT_CODE = 0
   END_OBJECT = EXIT_DEPENDENCY
   OBJECT = PCF_ENTRY
      CLASS = 11
      LOGICAL_ID = 600000
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MOD03"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = 0
      END_PERIOD_OFFSET = 0
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 12
      LOGICAL_ID = 700000
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MOD02QKM"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = 0
      END_PERIOD_OFFSET = 0
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 13
      LOGICAL_ID = 700002
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MOD021KM"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = 0
      END_PERIOD_OFFSET = 0
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
```

```
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 21
      LOGICAL_ID = 900000
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "GDAS_0ZF"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = -10650
      END_PERIOD_OFFSET = 10650
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 22
      LOGICAL_ID = 900020
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "OZ_DAILY"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = -43200
      END_PERIOD_OFFSET = 43200
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 23
      LOGICAL_ID = 900030
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "REYNSST"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = 0
```

```
      END_PERIOD_OFFSET = 0
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 24
      LOGICAL_ID = 900040
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "SEA_ICE"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = -43200
      END_PERIOD_OFFSET = 43200
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 25
      LOGICAL_ID = 900100
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "NISE"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = 0
      END_PERIOD_OFFSET = 0
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 31
      LOGICAL_ID = 420011
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MOD07LUT"
```

```
            DATA_TYPE_VERSION = "001"
            MIN_GRANULES_REQUIRED = 1
            MAX_GRANULES_REQUIRED = 1
            BEGIN_PERIOD_OFFSET = 0
            END_PERIOD_OFFSET = 0
            SCIENCE_GROUP = "L1"
            INPUT_TYPE = "Required"
            ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
            NUMBER_NEEDED = 1
            QUERY_TYPE = "Temporal"
            SPATIAL_TIME_DELTA = 0
            KEY_INPUT = "N"
            OBJECT = FILETYPE
                FILETYPE_NAME = "Single File Granule"
                CLASS = 1
            END_OBJECT = FILETYPE
        END_OBJECT = PCF_ENTRY
        OBJECT = PCF_ENTRY
            CLASS = 32
            LOGICAL_ID = 420012
            PCF_FILE_TYPE = 1
            DATA_TYPE_NAME = "MOD07LUT"
            DATA_TYPE_VERSION = "001"
            MIN_GRANULES_REQUIRED = 1
            MAX_GRANULES_REQUIRED = 1
            BEGIN_PERIOD_OFFSET = 0
            END_PERIOD_OFFSET = 0
            SCIENCE_GROUP = "L2"
            INPUT_TYPE = "Required"
            ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
            NUMBER_NEEDED = 1
            QUERY_TYPE = "Temporal"
            SPATIAL_TIME_DELTA = 0
            KEY_INPUT = "N"
            OBJECT = FILETYPE
                FILETYPE_NAME = "Single File Granule"
                CLASS = 1
            END_OBJECT = FILETYPE
        END_OBJECT = PCF_ENTRY
        OBJECT = PCF_ENTRY
            CLASS = 33
            LOGICAL_ID = 422501
            PCF_FILE_TYPE = 1
            DATA_TYPE_NAME = "MOD35ANC"
            DATA_TYPE_VERSION = "001"
            MIN_GRANULES_REQUIRED = 1
            MAX_GRANULES_REQUIRED = 1
            BEGIN_PERIOD_OFFSET = 0
            END_PERIOD_OFFSET = 0
            SCIENCE_GROUP = "L1"
            INPUT_TYPE = "Required"
            ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
            NUMBER_NEEDED = 1
            QUERY_TYPE = "Temporal"
            SPATIAL_TIME_DELTA = 0
            KEY_INPUT = "N"
            OBJECT = FILETYPE
                FILETYPE_NAME = "Single File Granule"
                CLASS = 1
```

```
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 34
      LOGICAL_ID = 900600
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MOD35ANC"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = 0
      END_PERIOD_OFFSET = 0
      SCIENCE_GROUP = "L2"
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 35
      LOGICAL_ID = 900601
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "MOD35ANC"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      BEGIN_PERIOD_OFFSET = 0
      END_PERIOD_OFFSET = 0
      SCIENCE_GROUP = "L3"
      INPUT_TYPE = "Required"
      ALIGN_DPR_TIME_WITH_INPUT_TIME = "N"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      SPATIAL_TIME_DELTA = 0
      KEY_INPUT = "N"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 122
      LOGICAL_ID = 402500
      PCF_FILE_TYPE = 2
      DATA_TYPE_NAME = "MODVOLC"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULE_YIELD = 1
      MAX_GRANULE_YIELD = 1
      ASSOCIATED_MCF_ID = 402503
      SCIENCE_GROUP = "S1"
      INSTANCE = 0
      MINIMUM_SIZE = 0
      MAXIMUM_SIZE = 0
```

```
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 123
      LOGICAL_ID = 420000
      PCF_FILE_TYPE = 2
      DATA_TYPE_NAME = "MOD07_L2"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULE_YIELD = 1
      MAX_GRANULE_YIELD = 1
      ASSOCIATED_MCF_ID = 420001
      SCIENCE_GROUP = "S2"
      INSTANCE = 0
      MINIMUM_SIZE = 0
      MAXIMUM_SIZE = 0
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 124
      LOGICAL_ID = 420002
      PCF_FILE_TYPE = 2
      DATA_TYPE_NAME = "MOD07_QC"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULE_YIELD = 0
      MAX_GRANULE_YIELD = 1
      ASSOCIATED_MCF_ID = 420003
      SCIENCE_GROUP = "S3"
      INSTANCE = 0
      MINIMUM_SIZE = 0
      MAXIMUM_SIZE = 0
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 125
      LOGICAL_ID = 422500
      PCF_FILE_TYPE = 2
      DATA_TYPE_NAME = "MOD35_L2"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULE_YIELD = 1
      MAX_GRANULE_YIELD = 1
      ASSOCIATED_MCF_ID = 422506
      SCIENCE_GROUP = "S4"
      INSTANCE = 0
      MINIMUM_SIZE = 0
      MAXIMUM_SIZE = 0
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY
```

```
OBJECT = PCF_ENTRY
   CLASS = 126
   LOGICAL_ID = 422551
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MOD35_QC"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 422507
   SCIENCE_GROUP = "S5"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 127
   LOGICAL_ID = 422552
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "MODCSR_G"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 422510
   SCIENCE_GROUP = "S6"
   INSTANCE = 0
   MINIMUM_SIZE = 0
   MAXIMUM_SIZE = 0
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 129
   LOGICAL_ID = 800510
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "SatelliteInstrument"
   PGE_PARAMETER_DEFAULT = "AM1M"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 130
   LOGICAL_ID = 402502
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "RP Reference to VOLCALERT"
   PGE_PARAMETER_DEFAULT = "402500:1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 131
   LOGICAL_ID = 420004
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "MOD_PR07.qc"
```

```
            PGE_PARAMETER_DEFAULT = "420002:1"
            PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
            PROFILE_SELECTOR_PGE_PARAMETER = "N"
    END_OBJECT = PCF_ENTRY
    OBJECT = PCF_ENTRY
            CLASS = 132
            LOGICAL_ID = 421000
            PCF_FILE_TYPE = 5
            PGE_PARAMETER_NAME = "Number_Of_Invent_RP"
            PGE_PARAMETER_DEFAULT = "4"
            PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
            PROFILE_SELECTOR_PGE_PARAMETER = "N"
    END_OBJECT = PCF_ENTRY
    OBJECT = PCF_ENTRY
            CLASS = 133
            LOGICAL_ID = 421001
            PCF_FILE_TYPE = 5
            PGE_PARAMETER_NAME = "Inventory_RP_Name_1 "
            PGE_PARAMETER_DEFAULT = "REPROCESSINGACTUAL"
            PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
            PROFILE_SELECTOR_PGE_PARAMETER = "N"
    END_OBJECT = PCF_ENTRY
    OBJECT = PCF_ENTRY
            CLASS = 134
            LOGICAL_ID = 421002
            PCF_FILE_TYPE = 5
            PGE_PARAMETER_NAME = "Inventory_RP_Value_1"
            PGE_PARAMETER_DEFAULT = "processed once"
            PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
            PROFILE_SELECTOR_PGE_PARAMETER = "N"
    END_OBJECT = PCF_ENTRY
    OBJECT = PCF_ENTRY
            CLASS = 135
            LOGICAL_ID = 421003
            PCF_FILE_TYPE = 5
            PGE_PARAMETER_NAME = "Inventory_RP_Name_2 "
            PGE_PARAMETER_DEFAULT = "REPROCESSINGPLANNED"
            PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
            PROFILE_SELECTOR_PGE_PARAMETER = "N"
    END_OBJECT = PCF_ENTRY
    OBJECT = PCF_ENTRY
            CLASS = 136
            LOGICAL_ID = 421004
            PCF_FILE_TYPE = 5
            PGE_PARAMETER_NAME = "Inventory_RP_Value_2"
            PGE_PARAMETER_DEFAULT = "further update is anticipated"
            PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
            PROFILE_SELECTOR_PGE_PARAMETER = "N"
    END_OBJECT = PCF_ENTRY
    OBJECT = PCF_ENTRY
            CLASS = 137
            LOGICAL_ID = 421005
            PCF_FILE_TYPE = 5
            PGE_PARAMETER_NAME = "Inventory_RP_Name_3 "
            PGE_PARAMETER_DEFAULT = "LOCALVERSIONID"
            PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
            PROFILE_SELECTOR_PGE_PARAMETER = "N"
    END_OBJECT = PCF_ENTRY
    OBJECT = PCF_ENTRY
```

```
      CLASS = 138
      LOGICAL_ID = 421006
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Inventory_RP_Value_3"
      PGE_PARAMETER_DEFAULT = "002"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 139
      LOGICAL_ID = 421007
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Inventory_RP_Name_4 "
      PGE_PARAMETER_DEFAULT = "PGEVERSION"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 140
      LOGICAL_ID = 421008
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Inventory_RP_Value_4"
      PGE_PARAMETER_DEFAULT = "3.0.0"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 141
      LOGICAL_ID = 421100
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Number_Of_Archive_RP"
      PGE_PARAMETER_DEFAULT = "8"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 142
      LOGICAL_ID = 421101
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Archive RP_Name_1 "
      PGE_PARAMETER_DEFAULT = "ALGORITHMPACKAGEACCEPTANCEDATE"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 143
      LOGICAL_ID = 421102
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Archive_RP_Value_1"
      PGE_PARAMETER_DEFAULT = "June 1997"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 144
      LOGICAL_ID = 421103
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Archive RP_Name_2 "
      PGE_PARAMETER_DEFAULT = "ALGORITHMPACKAGEMATURITYCODE"
```

```
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 145
   LOGICAL_ID = 421104
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive_RP_Value_2"
   PGE_PARAMETER_DEFAULT = "at-launch"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 146
   LOGICAL_ID = 421105
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive RP_Name_3 "
   PGE_PARAMETER_DEFAULT = "ALGORITHMPACKAGENAME"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 147
   LOGICAL_ID = 421106
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive_RP_Value_3"
   PGE_PARAMETER_DEFAULT = "ATBD-MOD-07"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 148
   LOGICAL_ID = 421107
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive RP_Name_4 "
   PGE_PARAMETER_DEFAULT = "ALGORITHMPACKAGEVERSION"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 149
   LOGICAL_ID = 421108
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive_RP_Value_4"
   PGE_PARAMETER_DEFAULT = "2"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 150
   LOGICAL_ID = 421109
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive RP_Name_5 "
   PGE_PARAMETER_DEFAULT = "INSTRUMENTNAME"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 151
```

```
          LOGICAL_ID = 421110
          PCF_FILE_TYPE = 5
          PGE_PARAMETER_NAME = "Archive_RP_Value_5"
          PGE_PARAMETER_DEFAULT = "Moderate Resolution Imaging Spectroradiometer"
          PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
          PROFILE_SELECTOR_PGE_PARAMETER = "N"
       END_OBJECT = PCF_ENTRY
       OBJECT = PCF_ENTRY
          CLASS = 152
          LOGICAL_ID = 421111
          PCF_FILE_TYPE = 5
          PGE_PARAMETER_NAME = "Archive RP_Name_6 "
          PGE_PARAMETER_DEFAULT = "Profiles_Algorithm_Version_Number"
          PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
          PROFILE_SELECTOR_PGE_PARAMETER = "N"
       END_OBJECT = PCF_ENTRY
       OBJECT = PCF_ENTRY
          CLASS = 153
          LOGICAL_ID = 421112
          PCF_FILE_TYPE = 5
          PGE_PARAMETER_NAME = "Archive_RP_Value_6"
          PGE_PARAMETER_DEFAULT = "1"
          PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
          PROFILE_SELECTOR_PGE_PARAMETER = "N"
       END_OBJECT = PCF_ENTRY
       OBJECT = PCF_ENTRY
          CLASS = 154
          LOGICAL_ID = 421113
          PCF_FILE_TYPE = 5
          PGE_PARAMETER_NAME = "Archive RP_Name_7 "
          PGE_PARAMETER_DEFAULT = "Total_Ozone_Algorithm_Version_Number"
          PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
          PROFILE_SELECTOR_PGE_PARAMETER = "N"
       END_OBJECT = PCF_ENTRY
       OBJECT = PCF_ENTRY
          CLASS = 155
          LOGICAL_ID = 421114
          PCF_FILE_TYPE = 5
          PGE_PARAMETER_NAME = "Archive_RP_Value_7"
          PGE_PARAMETER_DEFAULT = "1"
          PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
          PROFILE_SELECTOR_PGE_PARAMETER = "N"
       END_OBJECT = PCF_ENTRY
       OBJECT = PCF_ENTRY
          CLASS = 156
          LOGICAL_ID = 421115
          PCF_FILE_TYPE = 5
          PGE_PARAMETER_NAME = "Archive RP_Name_8 "
          PGE_PARAMETER_DEFAULT = "Stability_Indices_Algorithm_Version_Number"
          PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
          PROFILE_SELECTOR_PGE_PARAMETER = "N"
       END_OBJECT = PCF_ENTRY
       OBJECT = PCF_ENTRY
          CLASS = 157
          LOGICAL_ID = 421116
          PCF_FILE_TYPE = 5
          PGE_PARAMETER_NAME = "Archive_RP_Value_8"
          PGE_PARAMETER_DEFAULT = "1"
          PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
```

```
         PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 158
      LOGICAL_ID = 422508
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "MOD35_QC.qc"
      PGE_PARAMETER_DEFAULT = "422551:1"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 159
      LOGICAL_ID = 424000
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "MOD35_Num_InvMet_RP_Pairs"
      PGE_PARAMETER_DEFAULT = "4"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 160
      LOGICAL_ID = 424001
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Inventory_RP_Name_1 "
      PGE_PARAMETER_DEFAULT = "REPROCESSINGACTUAL"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 161
      LOGICAL_ID = 424002
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Inventory_RP_Value_1"
      PGE_PARAMETER_DEFAULT = "processed once"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 162
      LOGICAL_ID = 424003
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Inventory_RP_Name_2 "
      PGE_PARAMETER_DEFAULT = "REPROCESSINGPLANNED"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 163
      LOGICAL_ID = 424004
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Inventory_RP_Value_2"
      PGE_PARAMETER_DEFAULT = "further update is anticipated"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 164
      LOGICAL_ID = 424005
```

```
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Inventory_RP_Name_3 "
      PGE_PARAMETER_DEFAULT = "LOCALVERSIONID"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 165
   LOGICAL_ID = 424006
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Inventory_RP_Value_3"
   PGE_PARAMETER_DEFAULT = "002"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 166
   LOGICAL_ID = 424007
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Inventory_RP_Name_4 "
   PGE_PARAMETER_DEFAULT = "PGEVERSION"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 167
   LOGICAL_ID = 424008
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Inventory_RP_Value_4"
   PGE_PARAMETER_DEFAULT = "2.6.1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 168
   LOGICAL_ID = 424100
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "MOD35_Num_ArchMet_RP_Pairs"
   PGE_PARAMETER_DEFAULT = "5"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 169
   LOGICAL_ID = 424101
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive RP_Name_1 "
   PGE_PARAMETER_DEFAULT = "ALGORITHMPACKAGEACCEPTANCEDATE"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 170
   LOGICAL_ID = 424102
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive_RP_Value_1"
   PGE_PARAMETER_DEFAULT = "June 1997"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
```

```
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 171
   LOGICAL_ID = 424103
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive RP_Name_2 "
   PGE_PARAMETER_DEFAULT = "ALGORITHMPACKAGEMATURITYCODE"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 172
   LOGICAL_ID = 424104
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive_RP_Value_2"
   PGE_PARAMETER_DEFAULT = "at-launch"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 173
   LOGICAL_ID = 424105
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive RP_Name_3 "
   PGE_PARAMETER_DEFAULT = "ALGORITHMPACKAGENAME"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 174
   LOGICAL_ID = 424106
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive_RP_Value_3"
   PGE_PARAMETER_DEFAULT = "ATBD-MOD-06"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 175
   LOGICAL_ID = 424107
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive RP_Name_4 "
   PGE_PARAMETER_DEFAULT = "ALGORITHMPACKAGEVERSION"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 176
   LOGICAL_ID = 424108
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Archive_RP_Value_4"
   PGE_PARAMETER_DEFAULT = "2"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
   PROFILE_SELECTOR_PGE_PARAMETER = "N"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 177
   LOGICAL_ID = 424109
   PCF_FILE_TYPE = 5
```

```
      PGE_PARAMETER_NAME = "Archive RP_Name_5 "
      PGE_PARAMETER_DEFAULT = "INSTRUMENTNAME"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 178
      LOGICAL_ID = 424110
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Archive_RP_Value_5"
      PGE_PARAMETER_DEFAULT = "Moderate Resolution Imaging Spectroradiometer"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 179
      LOGICAL_ID = 424300
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "UW DEBUG; 0 to 4, no output to reams"
      PGE_PARAMETER_DEFAULT = "0"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 180
      LOGICAL_ID = 424301
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Processing Range Begin Line"
      PGE_PARAMETER_DEFAULT = "0"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 181
      LOGICAL_ID = 424302
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Processing Range Number of Lines"
      PGE_PARAMETER_DEFAULT = "0"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 182
      LOGICAL_ID = 424303
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Processing Range Begin Element"
      PGE_PARAMETER_DEFAULT = "0"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
   END_OBJECT = PCF_ENTRY
   OBJECT = PCF_ENTRY
      CLASS = 183
      LOGICAL_ID = 424304
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Processing Range Number of Elements"
      PGE_PARAMETER_DEFAULT = "0"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
      PROFILE_SELECTOR_PGE_PARAMETER = "N"
   END_OBJECT = PCF_ENTRY
```

END

A.4.8 GDAS_0ZF ODL

```
DATA_TYPE_NAME = "GDAS_0ZF"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "MODIS"
PLATFORM = "EOSAM1"
DATA_TYPE_DESCRIPTION = "NCEP 6-Hour Atmospheric Profile"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 21.0
PROCESSING_LEVEL = "L1"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "NCEP"
PERIOD = "HOURS=6"
BOUNDARY = "START_OF_6HOUR"
DURATION = "SECS=1"
DELAY = 10
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT = "GSFC"
PROCESSED_AT = "GSFC"
END
```

A.4.9 OZ_DAILY ODL

The same as GDAS_0ZF ODL except the following items:

```
DATA_TYPE_NAME = "OZ_DAILY"
DATA_TYPE_DESCRIPTION = "TOVS Column Ozone Daily Product"
NOMINAL_SIZE = 0.10
PERIOD = "DAYS=1"
BOUNDARY = "START_OF_DAY+43200"
DURATION = "SECS=1"
```

A.4.10 REYNSST ODL

The same as GDAS_0ZF ODL except the following items:

```
DATA_TYPE_NAME = "REYNSST"
DATA_TYPE_DESCRIPTION = "Reynolds Weekly SST"
NOMINAL_SIZE = 0.30
PERIOD = "SECS=604800"
BOUNDARY = "START_OF_WEEK-86400"
DURATION = "SECS=604800"
```

A.4.11 SEA_ICE ODL

The same as GDAS_0ZF ODL except the following items:

```
DATA_TYPE_NAME = "SEA_ICE"
DATA_TYPE_DESCRIPTION = "NCEP Ice Concentration"
```

```
NOMINAL_SIZE = 0.30
PERIOD = "SECS=86400"
BOUNDARY = "START_OF_DAY"
DURATION = "SECS=1"
```

A.4.12 NISE ODL

The same as GDAS_0ZF ODL except the following items:

```
DATA_TYPE_NAME = "NISE"
DATA_TYPE_DESCRIPTION = "NSIDC NISE snow/ice extent"
NOMINAL_SIZE = 0.03
PERIOD = "DAYS=1"
BOUNDARY = "START_OF_DAY"
DURATION = "DAYS=1"
```

A.5 Typical AIRS PGE & ESDT ODL Files

Listings are provided for the following AIRS ODL files:

A.5.1 AIRS PGE ODL for PGE_NAME AiL1A_AMSU

A.5.2 AIRS ESDT AIR10SCI ODL

A.5.3 AIRS ESDT AIR10SCC ODL

A.5.4 AIRS ESDT AIR20SCI ODL

A.5.5 AIRS ESDT PMCO_HK ODL

A.5.6 AIRS ESDT PM1EPHND ODL

A.5.7 AIRS ESDT PM1ATTNR ODL

A.5.8 AIRS ESDT AIRAASCI ODL

A.5.9 AIRS ESDT AIRXADCM ODL

A.5.10 AIRS ESDT AIRXATCM ODL

A.5.11 AIRS ESDT AIRXATCS ODL

A.5.12 AIRS ESDT AIRXARYL ODL

A.5.13 AIRS ESDT AIRXAGEO ODL


```
A typical AIRS PGE will differ from the examples here by the PGE_NAME, the
specific input/output files referenced, and runtime parameters.  However, the
overall structure of a given AIRS PGE ODL file would be similar to the ones
used here. (N.B. The ODL files shown here are associated with the AIRS version
2.1.2 software.)
```

A.5.1 AIRS PGE AiL1A_AMSU ODL
```
PGE_NAME = "L1A_AMSU"
PGE_VERSION = "212"
```

```
PROFILE_ID = 1
PROFILE_DESCRIPTION = "GRAN01"
PLATFORM = "EOSPM1"
INSTRUMENT = "AIRS"
MINIMUM_OUTPUTS = 0
SCHEDULE_TYPE = "Time"
PROCESSING_PERIOD = "MINS=6"
PROCESSING_BOUNDARY = "START_OF_DAY-31"
PGE_SSW_VERSION = "212"

/******* Primary Inputs *******/
OBJECT = PCF_ENTRY
   CLASS = 11
   LOGICAL_ID = 261
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "AIR10SCC"
   DATA_TYPE_VERSION = "001"
   BEGIN_PERIOD_OFFSET = 31
   END_PERIOD_OFFSET = 31
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   INPUT_TYPE = "Required"
/*   ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y"   */
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 12
   LOGICAL_ID = 262
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "AIR10SCI"
   DATA_TYPE_VERSION = "001"
   BEGIN_PERIOD_OFFSET = 31
   END_PERIOD_OFFSET = 31
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   INPUT_TYPE = "Required"
/*   ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y" */
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   OBJECT = FILETYPE
      FILETYPE_NAME = "L0 Data Files"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 13
   LOGICAL_ID = 290
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "AIR20SCI"
   DATA_TYPE_VERSION = "001"
   BEGIN_PERIOD_OFFSET = 31
   END_PERIOD_OFFSET = 31
```

```
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      INPUT_TYPE = "Required"
/*    ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y" */
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      OBJECT = FILETYPE
         FILETYPE_NAME = "L0 Data Files"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

/********Dynamic ancillary inputs *******/
OBJECT = PCF_ENTRY
      CLASS = 14
      LOGICAL_ID = 4007
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "PMCO_HK"
      DATA_TYPE_VERSION = "001"
      BEGIN_PERIOD_OFFSET = 31
      END_PERIOD_OFFSET = 31
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      INPUT_TYPE = "Required"
      /* ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y" */
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
      CLASS = 14
      LOGICAL_ID = 4008
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "PMCO_HK"
      DATA_TYPE_VERSION = "001"
      BEGIN_PERIOD_OFFSET = 31
      END_PERIOD_OFFSET = 31
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      INPUT_TYPE = "Required"
      /* ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y" */
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

/**** Attitude/Ephemeris/DEM entry.  Please delete if not used by PGE. ****/
OBJECT = PCF_ENTRY
      CLASS = 18
      LOGICAL_ID = 10501
      PCF_FILE_TYPE = 1
      DATA_TYPE_NAME = "PM1EPHND"
```

```
   DATA_TYPE_VERSION = "001"
   BEGIN_PERIOD_OFFSET = 31
   END_PERIOD_OFFSET = 31
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 19
   LOGICAL_ID = 10502
   PCF_FILE_TYPE = 1
   DATA_TYPE_NAME = "PM1ATTNR"
   DATA_TYPE_VERSION = "001"
   BEGIN_PERIOD_OFFSET = 31
   END_PERIOD_OFFSET = 31
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

/******************Primary output ************/
OBJECT = PCF_ENTRY
   CLASS = 110
   LOGICAL_ID = 7120
   PCF_FILE_TYPE = 2
   DATA_TYPE_NAME = "AIRAASCI"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULE_YIELD = 1
   MAX_GRANULE_YIELD = 1
   ASSOCIATED_MCF_ID = 17120
   SCIENCE_GROUP = "S1"
   MINIMUM_SIZE = 1
   MAXIMUM_SIZE = 100
   OBJECT = FILETYPE
      FILETYPE_NAME = "Single File Granule"
      CLASS = 1
   END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

/*************Static ancillary inputs *************/
OBJECT = PCF_ENTRY
   CLASS = 116
   LOGICAL_ID = 4001
   PCF_FILE_TYPE = 3
   DATA_TYPE_NAME = "AIRXADCM"
   DATA_TYPE_VERSION = "001"
```

```
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      SCIENCE_GROUP = "O001"
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY

   OBJECT = PCF_ENTRY
      CLASS = 117
      LOGICAL_ID = 4002
      PCF_FILE_TYPE = 3
      DATA_TYPE_NAME = "AIRXATCM"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      SCIENCE_GROUP = "O002"
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY

   OBJECT = PCF_ENTRY
      CLASS = 118
      LOGICAL_ID = 4003
      PCF_FILE_TYPE = 3
      DATA_TYPE_NAME = "AIRXATCS"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      SCIENCE_GROUP = "O003"
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
      QUERY_TYPE = "Temporal"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
   END_OBJECT = PCF_ENTRY

   OBJECT = PCF_ENTRY
      CLASS = 119
      LOGICAL_ID = 4005
      PCF_FILE_TYPE = 3
      DATA_TYPE_NAME = "AIRXARYL"
      DATA_TYPE_VERSION = "001"
      MIN_GRANULES_REQUIRED = 1
      MAX_GRANULES_REQUIRED = 1
      SCIENCE_GROUP = "O004"
      INPUT_TYPE = "Required"
      NUMBER_NEEDED = 1
```

```
      QUERY_TYPE = "Temporal"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 120
   LOGICAL_ID = 4006
   PCF_FILE_TYPE = 3
   DATA_TYPE_NAME = "AIRXAGEO"
   DATA_TYPE_VERSION = "001"
   MIN_GRANULES_REQUIRED = 1
   MAX_GRANULES_REQUIRED = 1
   SCIENCE_GROUP = "O005"
   INPUT_TYPE = "Required"
   NUMBER_NEEDED = 1
   QUERY_TYPE = "Temporal"
      OBJECT = FILETYPE
         FILETYPE_NAME = "Single File Granule"
         CLASS = 1
      END_OBJECT = FILETYPE
END_OBJECT = PCF_ENTRY

OBJECT = PCF_ENTRY
   CLASS = 125
   LOGICAL_ID = 1001
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Instrument: 0=AMSU, 1=AIRS, 2=HSB(MHS), 3=VIS"
   PGE_PARAMETER_DEFAULT = "0"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 225
   LOGICAL_ID = 1002
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Print Level IO: 0=Off, 1=Low, 2=Med, 3=High"
   PGE_PARAMETER_DEFAULT = "2"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 226
   LOGICAL_ID = 1003
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Print Level: 0=Off, 1=Low, 2=Med, 3=High"
   PGE_PARAMETER_DEFAULT = "1"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 227
   LOGICAL_ID = 1004
   PCF_FILE_TYPE = 5
   PGE_PARAMETER_NAME = "Exec Development Mode: 0=Off, 1=On"
   PGE_PARAMETER_DEFAULT = "0"
   PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
   CLASS = 228
```

```
      LOGICAL_ID = 1005
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Run Level 2 Mode: 1=MIT & 2=NOAA & 4=GSFC"
      PGE_PARAMETER_DEFAULT = "7"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 126
      LOGICAL_ID = 1006
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Stats Mode: 0=Off, 1=cmp2truth, 2=cmp2MW-retrieval"
      PGE_PARAMETER_DEFAULT = "0"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 128
      LOGICAL_ID = 1011
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Year (ex: 1998)"
      PGE_PARAMETER_DEFAULT = "1998"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 129
      LOGICAL_ID = 1012
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Month number (1 - 12)"
      PGE_PARAMETER_DEFAULT = "09"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 130
      LOGICAL_ID = 1013
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Day of month (1 - 31)"
      PGE_PARAMETER_DEFAULT = "13"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 131
      LOGICAL_ID = 1014
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Orbit of day (1 - 17)"
      PGE_PARAMETER_DEFAULT = "1"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 132
      LOGICAL_ID = 1015
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Granule Number (1 - 17)"
      PGE_PARAMETER_DEFAULT = "01"
      PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
      CLASS = 133
      LOGICAL_ID = 1016
      PCF_FILE_TYPE = 5
      PGE_PARAMETER_NAME = "Granule Size in scansets (1 - 45)"
```

```
     PGE_PARAMETER_DEFAULT = "45"
     PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
     CLASS = 134
     LOGICAL_ID = 1020
     PCF_FILE_TYPE = 5
     PGE_PARAMETER_NAME = "Times Processed: 1 for never before reprocessed"
     PGE_PARAMETER_DEFAULT = "1"
     PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
     CLASS = 135
     LOGICAL_ID = 1021
     PCF_FILE_TYPE = 5
     PGE_PARAMETER_NAME = "Processing Facility: A for TLSCF or G for GDAAC"
     PGE_PARAMETER_DEFAULT = "G"
     PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
OBJECT = PCF_ENTRY
     CLASS = 200
     LOGICAL_ID = 411
     PCF_FILE_TYPE = 5
     PGE_PARAMETER_NAME = "GDAAC Build Version String"
     PGE_PARAMETER_DEFAULT = "PGE=2.1.2, SDPTK=5.2.7.2, HDF=4.1r3,HDFEOS=2.7,
OS=6.5, COMPILER=7.2.1.3, ECS=6A.03"
     PGE_PARAMETER_DYNAMIC_VALUE = "NONE"
END_OBJECT = PCF_ENTRY
END
```

A.5.2 AIRS ESDT AIR10SCI ODL

```
DATA_TYPE_NAME = "AIR10SCI"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "AMSU_A1 Science Data Packets"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = .02
PROCESSING_LEVEL = "L0"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "GSFC"
PERIOD = "HOURS=2"
BOUNDARY = "START_OF_DAY"
SPATIAL_FLAG  = "N"
DELAY = 43200
OBJECT = USE_OBJECT
  CLASS = 1
  USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT   = "GSFC"
PROCESSED_AT  = "GSFC"
OBJECT = FILETYPE
     CLASS = 1
     FILETYPE_NAME = "L0 Data Files"
     MAXIMUM_NUM_FILES = 2
END_OBJECT = FILETYPE
```

```
END
```

A.5.3 AIRS ESDT AIR10SCC ODL

```
DATA_TYPE_NAME = "AIR10SCC"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "AMSU_A1 Science Data Packets"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = .02
PROCESSING_LEVEL = "L0"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "GSFC"
PERIOD = "HOURS=2"
BOUNDARY = "START_OF_DAY"
SPATIAL_FLAG  = "N"
DELAY = 43200
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
OBJECT = FILETYPE
    CLASS = 1
    FILETYPE_NAME = "L0 Data Files"
    MAXIMUM_NUM_FILES = 2
END_OBJECT = FILETYPE
END
```

A.5.4 AIRS ESDT AIR20SCI ODL

```
DATA_TYPE_NAME = "AIR20SCI"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "AMSU_A2 Science Data Packets"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = .02
PROCESSING_LEVEL = "L0"
HDF_DATA = "N"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "GSFC"
PERIOD = "HOURS=2"
BOUNDARY = "START_OF_DAY"
SPATIAL_FLAG  = "N"
DELAY = 43200
OBJECT = USE_OBJECT
    CLASS = 1
   USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
OBJECT = FILETYPE
    CLASS = 1
```

```
      FILETYPE_NAME = "L0 Data Files"
      MAXIMUM_NUM_FILES = 2
END_OBJECT = FILETYPE
END
```

A.5.5 AIRS ESDT PMCO_HK ODL

```
DATA_TYPE_NAME = "PMCO_HK"
DATA_TYPE_VERSION = "001"
DATA_TYPE_DESCRIPTION = "Aqua Carryout housekeeping file"
INSTRUMENT = "All"
PLATFORM = "Aqua"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 2.0
PROCESSING_LEVEL = "L0"
DYNAMIC_FLAG = "E"
PREDICTION_METHOD = "ROUTINE"
SUPPLIER_NAME = "GSFC"
PERIOD = "HOURS=2"
BOUNDARY = "START_OF_DAY"
DELAY = 43200
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
    USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
HDF_DATA = "N"
END
```

A.5.6 AIRS ESDT PM1EPHND ODL

```
DATA_TYPE_NAME = "PM1EPHND"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "All"
PLATFORM = "PM1"
DATA_TYPE_DESCRIPTION = "PM-1 FDD Definitive Ephemeris data in Toolkit format"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 6.0
PROCESSING_LEVEL = "L1"
DYNAMIC_FLAG = "I"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
    CLASS = 1
USED_BY = "GSFC"
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
HDF_DATA = "N"
END
```

A.5.7 AIRS ESDT PM1ATTNR ODL

```
DATA_TYPE_NAME = "PM1ATTNR"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "All"
PLATFORM = "PM1"
DATA_TYPE_DESCRIPTION = "PM-1 Refined Attitude Data in Toolkit format"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 1.0
```

```
PROCESSING_LEVEL = "L1"
DYNAMIC_FLAG = "I"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
HDF_DATA = "N"
END
```

A.5.8 AIRS ESDT AIRAASCI ODL

```
DATA_TYPE_NAME = "AIRAASCI"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "AMSU-A geolocated science counts"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 3.0
PROCESSING_LEVEL = "L1A"
HDF_DATA = "N"
PERIOD = "SECS=360"
BOUNDARY = "START_OF_SEC"
DYNAMIC_FLAG = "I"
SPATIAL_FLAG  = "N"
OBJECT = USE_OBJECT
   CLASS = 1
   USED_BY = "GSFC"
END_OBJECT = USE_OBJECT
ARCHIVED_AT  = "GSFC"
PROCESSED_AT  = "GSFC"
END
```

A.5.9 AIRS ESDT AIRXADCM ODL

```
DATA_TYPE_NAME = "AIRXADCM"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "Decommutation map"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 1.0
PROCESSING_LEVEL = "L1A"
HDF_DATA = "N"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
END
```

A.5.10 AIRS ESDT AIRXATCM ODL

```
DATA_TYPE_NAME = "AIRXATCM"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "Conversion method file"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 1.0
PROCESSING_LEVEL = "L1A"
```

```
HDF_DATA = "N"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
END
```

A.5.11   AIRS ESDT AIRXATCS ODL

```
DATA_TYPE_NAME = "AIRXATCS"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "Constant sets"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 1.0
PROCESSING_LEVEL = "L1A"
HDF_DATA = "N"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
END
```

A.5.12   AIRS ESDT AIRXARYL ODL

```
DATA_TYPE_NAME = "AIRXARYL"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "Red Yellow limits"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 1.0
PROCESSING_LEVEL = "L1A"
HDF_DATA = "N"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
END
```

A.5.13   AIRS ESDT AIRXAGEO ODL

```
DATA_TYPE_NAME = "AIRXAGEO"
DATA_TYPE_VERSION = "001"
INSTRUMENT = "AIRS"
PLATFORM = "EOSPM1"
DATA_TYPE_DESCRIPTION = "L1A.geolocation.anc"
PROVIDER = "Goddard Space Flight Center"
NOMINAL_SIZE = 1.0
PROCESSING_LEVEL = "L1A"
HDF_DATA = "N"
DYNAMIC_FLAG = "S"
SPATIAL_FLAG  = "N"
END
```

A.6   Typical Aqua MODIS PGE ODL File

Listings are provided for the following MODIS ODL files:

A.6.1 ODL files for Aqua MODIS PGE

The Aqua PGE ODL files are similar to those of Terra PGEs. The main differences are DATA_TYPE_NAME for input and output granules:

```
                            Terra          Aqua
----------------------------------------------------------------
L0                          MOD000         MODPML0

Static input                MOD01LUT       MYD01LUT

                            MOD03LUT       MYD03LUT

                            MOD02LUT       MYD02LUT
                            MOD07LUT       MYD07LUT
                            MOD35ANC       MYD35ANC
Ephemeris                   AM1EPHN0       PM1EPHND
Attitude                    AM1ATTNF       PM1ATTNR
Products                    MOD01          MYD01
                            MOD03          MYD03
                            MOD02QKM       MYD02QKM
                            MOD02HKM       MYD02HKM
                            MOD021KM       MYD021KM
                            MOD02OBC       MYD02OBC
                            MODVOLC        MYDVOLC
                            MOD07_L2       MYD07_L2
                            MOD07_QC       MYD07_QC
                            MOD35_L2       MYD35_L2
                            MOD35_QC       MYD35_QC
                            MODCSR_G       MYDCSR_G
```

Abbreviations and Acronyms

A$_o$	Operational Availability
ACL	Access Control List
ACS	Automated Cartridge System
ACSLS	Automated Cartridge Storage Library System
ADC	Affiliated Data Center (NOAA)
ADIC	Advanced Digital Information Corporation
ADSERV	Advertising Service
AFM	Affected File Metadata
AFL	Affected File List
AI&T	Algorithm Integration and Test
AIT	Algorithm Integration Team
ALOG	Application Log
AMASS	Archival Management and Storage System
APC	Access/Process Coordinators
API	Application Programming Interface
APID	Applications Process Identifier
ASCII	American Standard Code for Information Interchange
ASTER	Advanced Spaceborne Thermal Emission and Reflection Radiometer
BBS	Bulletin Board System
CAP	Cartridge Access Port
CCB	Configuration Control Board (NASA Convention)
CCR	Configuration Change Request
CCRS	Canada Centre for Remote Sensing
CD	Compact Disk
CDE	Common Desktop Environment
CD-ROM	Compact Disk - Read Only Memory
CDR	Critical Design Review

CDRL	Contract Data Requirements List
CDS	Cell Directory Service
CHCI	Communications Hardware Configuration Item
CHUI	Character User Interface
CI	Configuration Item
CIDM	Client Interoperability and Data Management
CLI	Command Line Interface
CM	Configuration Management
CMA	CM Administrator
CN	Change Notice
CO	Contracting Officer
COTR	Contracting Officer's Technical Representative
COTS	Commercial Off-the-Shelf (hardware or software)
CPU	Central Processing Unit
CR	Change Request
CRM	Change Request Manager
CSCI	Computer Software Configuration Item
CSMS	Communications and Systems Management Segment (ECS)
CSR	Consent To Ship Review
CSS	Communications Subsystem
DAA	Data Availability Acknowledgment
DAAC	Distributed Active Archive Center
DADS	Data Archive and Distribution System
DAN	Data Availability Notice
DAO	Data Assimilation Office
DAP	Delivered Algorithm Package
DAR	Data Acquisition Request
DAS	Data Availability Schedule
DAT	Digital Audio Tape
DB	Database

DBA	Database Administrator
DBMS	Database Management System
DBO	Database
DCE	Distributed Computing Environment (OSF)
DCF	Data Capture Facility
DCN	Document Change Notice
DCO	Document Change Order
DCR	Data Collection Request
DD	Data Dictionary
DDA	Data Delivery Acknowledgment
DDICT	Data Dictionary
DDIST	Data Distribution
DDL	Data Definition Language
DDN	Data Delivery Notice
DDSRV	Document Data Server
DDTS	Distributed Defect Tracking System
DEM	Digital Elevation Models
DES	Data Encryption Standard
DESKT	Desktop Configuration Item
DFA	Deleted From Archive Deletion From Archive
DID	Data Item Description
DIF	Data Interchange Formant
DIMGR	Distributed Information Manager
DLT	Digital Linear Tapes
DME	Distributed Management Environment
DMO	Data Management Organization
DNS	Domain Name Service
DOF	Distributed Object Framework
DOY	Day of Year

DP	Data Pool
DPAD	Data Pool Action Dispatcher
DPASU	Data Pool Access Statistics Utility
DPIU	Data Pool Insert Utility
DPM	Data Pool Maintenance
DPREP	Data Preprocessing
DPR	Data Processing Request
DPS	Data Processing Subsystem
DR	Delivery Record
DS	Data Server
DSI	Database Server Interface7
DSS	Data Server Subsystem
e-mail	Electronic Mail
EBnet	EOSDIS Backbone Network (combines Ecom and ESN)
ECHO	EOS Clearing House
ECN	Equipment Control Number
ECS	EOSDIS Core System
EDC	EROS Data Center (DAAC)
EDF	ECS Development Facility
EDHS	ECS Data Handling System
EGS	EOS Ground System
EIN	Equipment Identification Number
EMC	Enterprise Monitoring and Coordination
EMD	ECS Maintenance and Development (Project)
EMSn	EOS Mission Support Network (formerly EBnet)
EOC	EOS Operations Center (ECS)
EOS	Earth Observing System
EOSDIS	Earth Observing System Data and Information System
EP	Evaluation Package
EROS	Earth Resources Observation System

ESD	Electrostatic Discharge
ESDIS	Earth Science Data and Information System (GSFC Code 505)
ESDT	Earth Science Data Type
ESOD	Earth Science On-line Directory
ET	Eastern (standard or daylight savings) Time
FDD	Flight Dynamics Division
FDDI	Fiber Distributed Data Interface
FIFO	First in First out
FORTRAN	FORmula TRANslation (computer language)
FOS	Flight Operations Segment (ECS)
FOT	Flight Operations Team
FSMS	File and Storage Management System
FTP	File Transfer Protocol
FTPD	File Transfer Protocol Daemon
GB	Gigabyte (10^9)
Gb	Gigabit (10^9)
GBps	Gigabytes per Second
Gbps	Gigabits per Second
GBAD	Ground Based Attitude Determination
GCDIS	Global Change Data and Information System
GCMD	Global Change Master Directory
GDS	Ground Data System
GFE	Government Furnished Equipment
GFP	Government Furnished Property
GID	Group IDs
GIGE	Gigabit Ethernet
GSFC	Goddard Space Flight Center
GUI	Graphical User Interface
H/W	Hardware
HDF	Hierarchical Data Format

HIPPI	High Performance Parallel Interface
HPOV	HP Open View
HSM	Hierarchical Storage Management
HTML	Hypertext Mark-Up Language
HWCI	Hardware Configuration Item
I&AT	Integration and Acceptance Test
I&T	Integration and Test
I&TT	Integration and Test Team
IATO	Independent Acceptance Test Organization
ICD	Interface Control Document
ICLHW	Ingest Client Hardware [configuration item]
ILM	Inventory, Logistics, and Maintenance
ILP	Integrated Logistics Plan
ILS	Integrated Logistics Support
ILSMT	ILS Management Team
ILSO	ILS Office
INGST	Ingest Services
INS	Ingest System
IOS	Interoperability Subsystem
IP	Internet Protocol
IQ	Intelligent Query and IQ Access
Ir1	Interim Release 1
ISDN	Integrated Services Digital Network
ISS	Internetworking Subsystem
ISQL	Interactive (Structured Query Language) SQL
IV&V	Independent Verification and Validation
JIL	Job Information Language
JPL	Jet Propulsion Laboratory (DAAC)
KB	Kilobyte (10^3)
Kb	Kilobit (10^3)

KBps	Kilobytes per Second
Kbps	Kilobits per Second
L-7	Landsat-7 (Landsat-7 for EDHS search)
L0	Level 0
L0-L4	Level 0 (zero) through Level 4 (use Level-0 through Level-4 for EDHS search)
LAN	Local Area Network
Landsat	Land Remote-Sensing Satellite
LaRC	Langley Research Center (DAAC)
LCU	Library Control Unit
LDOS	Landsat Data and Operations System
LIM	Local Information Manager
LIMGR	Local Information Manager
LMC	Local Maintenance
LMU	Library Management Unit
Loral	Loral Aerosys (ECS Team)
LRU	Line Replaceable Unit
LSM	Library Storage Module
LTM	Log Transfer Manager
LZPF	Level Zero Processing Facility
M	Million, mega (prefix)
M&O	Maintenance and Operations
MAN	Metropolitan Area Network
MB	Megabyte (10^6)
Mb	Megabit (10^6)
MBps	Megabytes per Second
Mbps	Megabits per Second
MCF	Metadata Configuration File Metadata Control File
MD	Master Directory
MDA	Management Data Access

611-EMD-001

MDT	Mean Downtime
MHWCI	Management Hardware Configuration Item
MHz	Megahertz
MIB	Management Information Base
MIS	Management Information System
MM	Millimeter
MO&DSD	Mission Operations and Data Systems Directorate (GSFC Code 500)
MOU	Memorandum of Understanding
MR	Malfunction Report
MSEC	Millisecond
MSFC	Marshall Space Flight Center (DAAC)
MSS	Systems Management Subsystem
MTBCM	Mean Time Between Corrective Maintenance
MTBF	Mean Time Between Failure
MTBM	Mean Time Between Maintenance
MTBPM	Mean Time Between Preventive Maintenance
MTPE	Mission to Planet Earth
MTTR	Mean Time to Repair
MTTRes	Mean Time to Restore
MTU	Maximum Transfer Unit
MWO	Maintenance Work Order
N/A	Not Applicable
NA	Network Administrator
NASA	National Aeronautics and Space Administration
Nascom	NASA Communications
NCC	Network Control Center (GSFC) network communication center
NCEP	National Centers for Environmental Prediction
NCR	Nonconformance Report
NCS	Netscape Commerce Server

NCSA	National Center for Supercomputer Applications
NMCI	Network Management Configuration Item
NNM	HPOpenView Network Node Manager
NOAA	National Oceanic and Atmospheric Administration
NPG	NASA Procedures and Guidelines
NSI	NASA Science Internet
NWCI	Networking Configuration Item
ODL	Object Description Language
OEM	Original Equipment Manufacturer
OI	Operator Interface
OJT	On-the-Job Training
OM	Order Manager
OPR	Operator
OPS	Operations Operations Supervisor
Ops Super	Operations Supervisor
ORPA	Operations Readiness & Performance Assurance
ORR	Operations Readiness Review
OS	Operating System
OSF	Open Software Foundation
OTS	Off the Shelf
OVW	HP OpenView Windows
PAIP	Performance Assurance Implementation Plan
PB	Petabyte (10^{15})
PC	Personal Computer Process Control
PCF	Process Control File
PDL	Program Design Language
PDPS	Planning and Data Processing System
PDR	Product Delivery Record

PDS	Product Distribution System Production Data Sets
PDSOI	Product Distribution System Operator Interface
PDSIS	Product Distribution System Interface Server
PGE	Product Generation Executable
PGS	Product Generation Service
PH	Production History
PI	Principal Investigator
PIN	Password Identification Number
PLANG	Production Planning CSCI
PLNHW	Planning Hardware [configuration item]
PLS	Planning Subsystem
PM	Preventative Maintenance
PMPDR	Physical Media Product Delivery Record
PPM	Principal Period of Maintenance
PR	Production Request (s)
PRE	Production Request Editor
PRS	Primary Replication Server
QA	Quality Assurance Quality Assessment
QAMUT	Quality Assessment Metadata Update Tool
QC	Quality Control
QRU	Query, Retrieve, and Update
R&M	Reliability and Maintainability
RAID	Redundant Array of Inexpensive Disks
RAM	Random Access Memory
RCL	Replication Command Language
RE	Responsible Engineer
RID	Review Item Discrepancy
RMA	Return Material Authorization
RMS	Resource Management Subsystem

RPC	Remote Procedure Call
RRS	Replicate Replication Server
RS	Replication Server
RSA	Replication System Administrator
RSM	Replication Server Manager
RSSD	Replication Server System Database
S/C	Spacecraft
S/W	Software
S/WCI	Software Configuration Item
SA	System Administrator
SATAN	Security Administrator Tool for Analyzing Networks
SCDO	Science and Communications Development Office (ECS)
SCF	Science Computing Facility
SCID	Space Craft ID
SCSI	Small Computer System Interface
SDP	Science Data Processing
SDPF	Science Data Processing Facility
SDPS	Science Data Processing Segment (ECS)
SDPS/W	Science Data Production Software
SDPTK	Science Data Processing Toolkit
SDSRV	Science Data Server
SE	System Engineering
SE&I	System Engineering and Integration
SEI&T	System Engineering, Integration, and Test
SEO	Sustaining Engineering Organization
SEPG	Software Engineering Process Group
SGI	Silicon Graphics Incorporated
SI&T	System Integration and Test
SLA	Site License Agreement Software License Administrator

SMC	System Monitoring and Coordination Center
SMF	Status Message Facility
SMTP	Simple Mail Transport Protocol
SNMP	Simple Network Management Protocol
SOR	System Operations Review
SORR	Segment Operational Readiness Review
SPRHW	Science Processing Hardware [configuration item]
SQL	Structured Query Language
SQR	SQL Report Writer
SQS	Spatial Query Server
SRR	System Requirements Review
SSAP	Science Software Archive Package
SSH	Secure Shell
SSI&T	Science Software Integration and Test
SSL	Secure Socket Layer
SSS	Spatial Subscription Server Secure Shell Setup
STK	Storage Tek
STMGT	Storage Management
SubsMgr	Subscription Manager
SUBSRV	Subscription Server
T&M	Time and Materials
TB	Terabyte (10^{12})
TBC	To Be Confirmed
TBD	To Be Determined
TBR	To Be Resolved
TBS	To Be Supplied
Tbyte	Terabyte
TCP/IP	Transmission Control Protocol/Internet Protocol
TEC	Tivoli Enterprise Console

611-EMD-001

telecon	Telephone Conference
TELNET	Telecommunication Network
TRMM	Tropical Rainfall Measurement Mission
TSDIS	TRMM Science Data and Information System
TT	Trouble Ticket
UDP	User Datagram Protocol
UID	User IDs
UR	Universal Reference
URDB	User Recommendations Database
URL	Universal Resource Locator
USO	User Support Office
US Rep	User Services Representative
UWG	User Working Group
VDD	Version Description Document
VOB	Versioned Object Base (ClearCase)
WAIS	Wide Area Information Server
WAN	Wide Area Network
WKBCH	Workbench
WKSHCI	Working Storage Hardware Configuration Item
WWW	World Wide Web
XLV	Logical Volume Disk Driver

This page intentionally left blank.